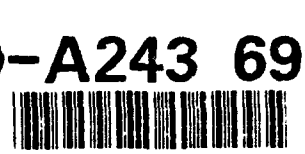


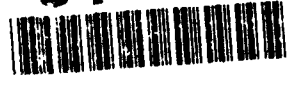
AFIT/GEO/ENG/91D-06

FEATURE EXTRACTION FOR POSE ESTIMATION

THESIS

Donald J. Willis
Captain, USAF

AFIT/GEO/ENG/91D-06



Approved for public release; distribution unlimited

REPORT DOCUMENTATION PAGE

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	December 1991	Master's Thesis

4. TITLE AND SUBTITLE

FEATURE EXTRACTION FOR POSE ESTIMATION: A COMPARISON BETWEEN SYNTHETIC AND REAL IR IMAGERY

5. FUNDING NUMBERS

6. AUTHOR(S)

Donald J. Willis, Capt, USAF

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Air Force Institute of Technology
WPAFB OH 45433-6583

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT/ENG/GEO/91D-06

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

WL/AARA
WPAFB OH 45433-6583

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION / AVAILABILITY STATEMENT

"Approved for Public Release: Distribution Unlimited."

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

This research addressed the problem of pose estimation of three-dimensional objects given their two-dimensional IR imagery and corresponding synthetic (computer-generated) IR imagery. Features and techniques were investigated to find those which may be extendable from computer models to real-world IR imagery. GTSIG and SCNGEN were used to create the synthetic imagery. Silhouette and outline shape moments were explored as optimum features for the comparison. Employing back-propagation with momentum as the training paradigm, a two-hidden-layer neural network was able to determine the base-plane orientation of the synthetic imagery to within 7.5 degrees with better than 90% accuracy. (No conclusive results were obtained from comparison with real-world IR imagery.) Additionally, the use of object hot spots relative to object height-to-width ratio is briefly discussed as an alternative feature/technique.

14. SUBJECT TERMS			15. NUMBER OF PAGES
Pose Estimation	**SCNGEN**		**175**
Synthetic IR Imagery	**Model Based**		16. PRICE CODE
GTSIG	**Orientation**		

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	**Unclassified**	**Unclassified**	**UL**

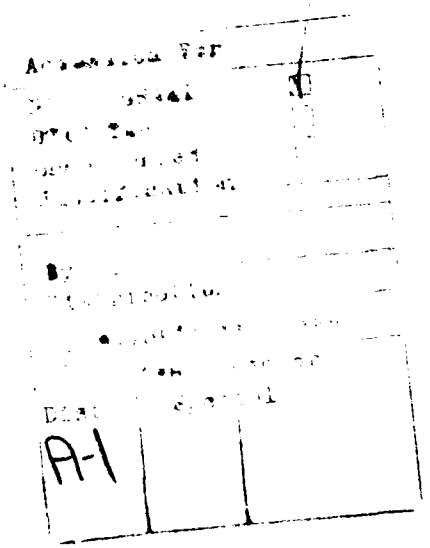
AFIT/GEO/ENG/91D-06

FEATURE EXTRACTION FOR POSE ESTIMATION: A COMPARISON BETWEEN SYNTHETIC AND REAL IR IMAGERY

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology
Air University
In Partial Fulfillment of the
Requirements for the Degree of
Master of Science in Electrical Engineering

Donald J. Willis, B.S., B.S.E.E.
Captain, USAF



December, 1991

Acknowledgments

I cannot overstate my appreciation for Major Steven K. Rogers (PhD) as my thesis advisor. He guided, prodded, and pulled me through the problems this research posed. He was always supportive, encouraging and focused.

Thanks to Dr Matt Kabrisky for his advice and recommendations. His retirement from AFIT will be an immeasurable loss to this institution. Thanks to Lt Col Phil Amburn and Capt Dennis Ruck. Never hesitant to help, they provided me with instructional information any time I asked.

Many thanks to the crucial support of the staff at the Model-Based Vision lab here at Wright-Patterson AFB. Specifically, Rick Beam's technical assistance in operating the computer systems and Bill Foley's and Kevin Wiley's support in operating the software and resolving software problems was invaluable to me. Thanks to Jim Leonard for his patience in showing me imagery and discussing LIDAR details with me.

Thanks to the Georgia Institute of Technology for its research grade IR modeling software (GTSIG and SCNGEN). Without Georgia Tech's software and assistance, this research would not have been possible.

Most importantly, I thank Susan L. Karch. Her keen mind, gentle manner, kind voice, and welcomed advice have always carried me through my toughest moments. I would have been lost without you, Susan.

Donald J. Willis

Table of Contents

List of Figures

List of Tables

Abstract

This research addresses the problem of pose estimation of three dimensional objects given their two-dimensional IR imagery and corresponding synthetic (i.e., computer-generated) IR imagery. In this research, features and techniques are investigated to find those which may be extendable from synthetic imagery to real-world imagery.

For creating the synthetic imagery, the computer programs GTSIG and SCNGEN were used. Based on the strengths and weaknesses of these computer programs, silhouette and outline shape moments were selected as optimum first choices for identifying the base-plane rotation angle of a Soviet T62 Main Battle Tank.

Employing "backpropagation with momentum" as the training paradigm, a neural network was used to identify the base-plane rotation angle of the synthetic T62 to within $\pm$ 7.5 degrees with an accuracy greater than 90%. However, no conclusive results could be drawn from comparisons to real IR imagery.

FEATURE EXTRACTION FOR POSE ESTIMATION: A COMPARISON BETWEEN SYNTHETIC AND REAL IR IMAGERY

I. Problem Description

1.1 Background

The Air Force Institute of Technology (AFIT) investigates the many aspects of three-dimensional object recognition using two-dimensional imagery. Much of the research in this area involves the recognition of strictly military objects of interest (such as tanks and jeeps—as opposed to chairs, tables, and so forth). As should be expected, these items can be targeted or used for strategic or tactical intelligence purposes.

One method of automatic recognition using computers involves the correlation of a reference object with a test object. The magnitude and shape of the correlation peak indicate the degree of correspondence between the reference and the test object. The location of the correlation peak within the test object's imagery also indicates where the object is located within that imagery.

For real life applications, accurate correlations require an enormous amount of reference images for the comparisons. Physical constraints such as memory size prohibit such a large data base of actual imagery. Additionally, to generate the imagery, a real object has to be rotated (and imaged) through every possible orientation angle (4π steradians worth of rotation—half this number if you are looking at objects on the ground since you generally can't see the undersides of the objects).

Consequently, the use of computer-generated (or synthetic) imagery for this correlation is an attractive alternative. The only memory required is that for the description of the

model's physical geometry and radiative properties. Any orientation of the model can be generated—giving an unlimited number of images for correlation purposes.

However, to generate synthetic imagery to use for the correlations, some estimation of the orientation of the real object is required. A sufficient number of orientation parameters must be determined in order to generate the synthetic image which closely matches the real object—a number which is dependent on the software used to generate the synthetic imagery.

Furthermore, features must be found which can be applied to both the synthetic and real imagery (and which can be used to determine the orientations of both). These features will be strongly dependent on the software used to generate the synthetic imagery. Depending on the modeling method used and the realism obtainable, features which work for one program's synthetic imagery may not work for another's.

In addition to the above use of correlation techniques for object recognition, neural networks have been studied extensively at AFIT. A neural network's ability to group and segregate large amounts of varying data cannot be overstated. In fact, theoretically, a two-layer multilayer-perceptron neural network can group and discriminate any number of items based solely on the number of nodes within each layer (71). As such, a neural network may be useful in sifting through and sorting out the numerical data obtained from the extraction of features from an image.

1.2 Problem

Complete recognition of real three-dimensional objects from two-dimensional imagery (i.e., regular pictures—as opposed to holograms) is an unsolved problem. Complete recognition requires identification of the object along with determination of its orientation (tilt, slant, and rotation angles—or yaw, pitch and roll angles in aeronautical terms). This thesis will investigate a solution to determining the orientation of objects from their two-dimensional imagery.

This research will attack this problem by using comparisons between synthetic (computer generated) and real infrared (IR) imagery. Therefore, this research will also address the problem of determining those features which yield orientation information for both synthetic and real IR imagery. Such features will be strongly dependent on the software used to generate the synthetic imagery; therefore, an assessment of the specific image-rendering software (GTSIG and SCNGEN) will be provided.

1.3 Summary of Current Knowledge

As will be shown in Chapter 2, there are many features and techniques being used to both recognize and determine the pose of three-dimensional objects given two-dimensional imagery. These features and their supporting techniques can be grouped according to the number of dimensions considered in the feature extraction or image processing stages:

- Volume-Based. Combines the two-dimensions of the image with a third dimensional quantity (depth or range) obtained through direct or indirect techniques.
- Surface-Based. Considers only the two-dimensions present in the image itself. Treats the object as a single flat entity or as a collection of coplanar regions.
- Line-Based. Detects and processes features based on edges or the junction of edges (vertices). These lines are conceptually one dimensional; however, in real practice, there is always a finite extend in the orthogonal direction.
- Point-Based. Uses "zero-dimensional" points found within an image. These points are obtained directly from "landmarks" in the image or from the processing of the higher dimensional features.

As discussed in the next chapter, all of these techniques are attractive for solving the problem at hand; however, limitations of the modeling software relative to the features and techniques will have to be considered. Models that appear visually precise may not be mathematically precise enough to support some of the techniques. Similarly, models

that are too precise in some regards, but not precise in others, may not be mathematically extendable to real imagery comparisons (for example, modeling the effects of the atmosphere and a sensor's limitations are perhaps more important than being able to mathematically model the small temperature differences between small adjacent panels on an object).

1.4 Assumptions

For this research, the following assumptions are made:

- The imagery has been successfully segmented. That is, the potential objects of interest have been segregated from their surrounding background in such a manner that the object can be isolated from everything else in the imagery.
- The type of target (T-62 tank, jeep, truck, etc.) is known.
- The range to each object is known (through the use of LASER RADAR or conventional RADAR).
- The orientation of the patch of ground upon which a target rests is known (i.e., the tilt and slant angles are known—possible using LASER RADAR).

1.5 Approach/Methodology

This research will attempt to use computers to automatically determine the orientation of a real object from two-dimensional infrared (IR) imagery (i.e., pictures). To accomplish this, measurements obtained from the real imagery will be analyzed in a neural network which has been trained to compute the orientation angles of a similar synthetic object (the synthetic object is generated using a computer model).

To do this, the following actions are required:

- Assessment of the computer modeling software. This assessment will attempt to determine the limitations of the software so as to allow a judicious selection of features for comparison between the synthetic IR imagery and real-world IR imagery.

- Selection of features which can be applied to both real and synthetic IR imagery. These features can be external features (those based on shape features such as silhouettes, outlines, etc.) internal features (such as shading, location of hot spots, etc.), or a combination of the two (Gabor features, Fourier features, etc.)
- Validation that the selected features can be used to determine the orientation of an object in synthetic IR imagery. This will be demonstrated by training a neural network to find the orientation angles of objects within the synthetic imagery.
- Validation that the selected features can be used to determine the orientation of objects in real IR imagery. Once the neural network has been trained using the synthetic imagery, the real imagery's measurements will be processed. The neural network's output will be the orientation of the synthetic image which most closely corresponds to objects in the real imagery.
- Comparison between the real object and the synthetic object at the orientation angles provided by the neural network. This comparison will be in the form of a correlation of the two images. The location, shape, and relative magnitude of the strongest correlation peak should provide a figure of merit for assessing the accuracy of the technique.

1.6 Materials and Equipment

The Georgia Institute of Technology infrared signature (GTSIG) and scene generation (SCNGEN) software will be used to generate the synthetic imagery. Feature-processing software will be developed within this research for special applications. All software will be run upon the MicroSPARC workstations at the Model-Based Vision (MBV) lab here at Wright-Patterson Air Force Base.

1.7 Schedule

The first step is to determine the strengths and weaknesses of the specific image rendering software used for synthetic-image generation. Based on this assessment, features will

be selected which appear to be most comparable between the synthetic IR imagery and real IR imagery.

Next, the features will be investigated with regard to the synthetic imagery. To assess whether the features can be used for determining the orientation of the synthetic imagery, the features will be extracted and plotted as the orientation of the object within the synthetic imagery is varied. From an analysis of these plots, a subset of features will be selected for further investigation.

The selected subset of features will be evaluated for use in determining the orientation of the synthetic imagery. The features will be extracted from objects at known orientations using the synthetic imagery. The extracted features will be used to train a neural network to recognize synthetic object orientation. Various configurations of the network will be explored until an optimal configuration is found.

Finally, if the selected subset of features can be used in pose estimation of the objects in the synthetic imagery, the features will be extracted from real-world IR imagery and processed using the trained neural network. The results from the neural network will be used to generate a synthetic image for visual comparison to, and mathematical correlation with, the original real-world imagery. The resulting figure of merit (location, shape and relative magnitude of the correlation peak) will be used to assess the accuracy of the techniques and the usefulness of the selected features.

1.8 Summary

This research will address the problem of pose estimation of three-dimensional objects given their two-dimensional IR imagery. This effort will be centered around the use of computer generated IR imagery. The applicability of certain features will be discussed both in relation to the pose estimation problem and to the limitations of the computer modeling software.

As for the remainder of this document, Chapter II identifies techniques used by other researchers for identifying and estimating the pose of three-dimensional objects from their two-dimensional imagery. Chapter III provides justification for the methods selected and provides details on the approach taken in this research. Chapter IV reports the results obtained using the features and techniques selected for this pose estimation research. Lastly, Chapter V provides conclusions and recommendations.

II. Literature Review

2.1 Introduction

This literature review focuses on finding the many features and techniques used in 1) recognizing three-dimensional objects from their two-dimensional imagery. 2) recognizing two-dimensional objects using their three-dimensional models, and 3) estimating the three-dimensional pose of an object from its two-dimensional imagery. From this base of knowledge, features and techniques will be sought which may be extended to the specific problem of pose estimation of two-dimensional infrared (IR) images generated using computer models of three-dimensional objects.

2.2 Current Methods

The current methods for the research described in Section 2.1 can be divided into four broad categories: measurements based on volume, measurements based on surfaces, measurements based on lines, and measurements based on points. As noted by Fan *et al.* (29), these four categories represent a hierarchy:

High-Level Description	Invariant to Viewing Angle	Volume Measurements
		Surface Measurements
Mid-Level Description	Tolerant of Small Changes in Viewing Angle	
		Line Measurements
Low-Level Description	Dependent on Viewing Angle	Point Measurements

As shown above, the volume measurements represent a high-level description of an object which is totally invariant to the angle from which the object is viewed. As will be shown below, true volume representations are not realistic in a real-world target recognition

scenario. As such, the volume measurements discussed below are not "purest" volume measurements, but are measurements based on the visible sides of an object. They are higher level than the surface measurements which will be described, but lower level than true volume measurements. Key in the volume measurements is some determination or estimation of the relationships of various parts of the object to one another in three-dimensions.

For the other three categories of measurements, surface measurements include operations on such things as facets, silhouettes, outlines (boundary edges), etc. Surface measurements can be thought of as dealing with two-dimensional entities. Line measurements include edge-detection types of operations with some manipulation of the edge data once obtained. Conceptually, line measurements deal with one-dimensional entities. And, finally, point measurements involve the relationships of various points on the object (these points are, essentially, zero-dimensional entities).

It should be noted that these measurements are not always performed independently of one another. Furthermore, many of the measurements are taken at a high level and then converted to a lower level for processing (e.g., after finding the lines in an image, the centroids of the lines are used as points for point-based recognition).

2.2.1 Volume Measurements. As noted above, these measurements are not "purest" volume measurements. This is due to the fact that these measurements are taken from only the visible side or sides of an object. Portions of the object which are occluded by other portions of the object, as well as the underside and backsides of the object, preclude a full measurement of the volumetric features of an object in a real-world scenario.

The volume measurements are based on some determination or estimation of the three-dimensional relationships among parts of the object. These three-dimensional relationships are determined 1) via direct range measurements (using LIDAR or RADAR), 2) via indirect estimations of the relative orientations of portions of the object using shading analysis, and 3) via correlation between multiple views of an object.

2.2.1.1 Range-Based. Range-based measurements, as used here, are obtained using direct range measurements (68, 74, 16, 27, 10, 29), or by using stereoscopic vision techniques (13, 60), or by using some other ranging technique (70, 62, 64, 22, 60). The data obtained from this additional dimension is used to 1) enhance standard mathematical processing (such as moment-based analysis), 2) decompose the object into its constituent orthographic projections, 3) detect planar patches, 4) detect high-interest points, or 5) compute volume-based aspect ratios. Each technique is discussed below.

Three-Dimensional Moments. Representative of this work is the work by Sadjadi (74) and Casasent (15). They extended invariant moments to include range information and confirmed that this additional information could be used to classify three-dimensional objects.

Primary-Silhouettes. Primary silhouettes are obtained via the orthographic projections of a three-dimensional object (i.e., project the object onto its x-y, x-z, and y-z planes) (85). Wang *et al.* (85) used the projected silhouettes of an object to match to a library of models. The techniques included calculation of the moments up through the second-order moments and calculation of the Fourier descriptors. To account for dilation effects, Wang states that one can use the ratio of the principal moments of the object to the principal moments of the object's enclosing cube (or parallelepiped).

Planar Patches. Once found, these features are processed according to three basic qualities: relative locations in three-dimensions (38, 39); orientations in three-dimensions (56, 40, 83, 38, 39); and, dihedral angles (48). Notable to the present research effort, however, Roggeman's work (68, 69) suggests that for the ranges involved in many military target-acquisition environments (1 km or so), accurate small-scale orientation measurements may be difficult to obtain due to noise in the range measurement process.

Control Points. These high-interest points are those which can be shown to correspond with a moderately high-degree of certainty between a model and specific instances of an object--such as for over-head photography (76). As used by Lie (51), these points were

generated by finding the range to specific points in an intensity image. The combination of range and intensity produced fast and robust results.

Three-Dimensional Aspect Ratio. This ratio, analogous to the two-dimensional height-to-width aspect ratio, is calculated by dividing the measured volume of an object by the area of its projection toward the viewer (85).

2.2.1.2 Shape from Shading. As opposed to the direct measurement of the range as is done in the previous subsection, "shape from shading" attempts to estimate the relative ranges and orientations among surfaces of an object using the relative changes in intensity among the surfaces (43, 70, 62, 64, 59, 60, 13, 51, 54). It is based on the assumption that the intensity viewed from an object represents regions of relatively uniform reflectance as from a regular texture (54) or from regions of similar orientation (60).

This "range" information is also used to find planar patches (70, 62, 64). However, as noted by Oshima (60), "...there is no guarantee that changes in light intensity correspond to geometric features."

2.2.1.3 Multiple Views. To overcome the need for range information, this class of volume measurements uses multiple views of an object taken from at least two different camera viewing angles. These multiple views can be used to find internal features (13, 56) or to enhance representations such as silhouettes (20). Though this technique does not rely on range information, as noted by Chien (20), these measurements can be tied to models if range information is available.

2.2.1.4 Extended Gaussian Image. The Extended Gaussian Image (EGI) is more a technique for using three-dimensional data than a means of gathering it. To generate an EGI, a three-dimensional object is decomposed into its component surfaces in three-dimensions as will be shown below. Within its limitations, the EGI represents a complete volumetric description of an object which can be obtained using models of the object

(42, 43, 46, 29). As such, it is a candidate technique for matching features between models and objects.

As explained by Horn (42, 43), the Gaussian sphere is a unit sphere upon which points are placed to correspond to surfaces on an object. The placement of these points is such that the normal to the sphere at the point corresponds to the normal of the surface being represented. If, however, instead of just a point, a magnitude (or mass) is used which corresponds to the area of the surface represented, one has the EGI.

The EGI is invariant to translation, and rotates in correspondence with rotations of the object it represents (42). In its use, Ikeuchi (46) used three-dimensional surfaces features and matched the visible pattern to its corresponding portion of the EGI. However, as noted by Fan *et al.* (29), the "EGI is sensitive to occlusion and is unique for convex objects only." Furthermore, segmenting an image so as to be able to use the EGI is difficult for multiple scene analysis (29) (this may not be applicable to the research in this thesis given the assumptions about the imagery as stated in Chapter 1).

2.2.2 Surface Measurements. As implied in the introductory paragraphs of this section, surface measurements do not use the third dimension of depth (or range). The resulting two-dimensional features treat the object as a single flat surface (referred to as "Overall Shape" below) or as a collection of flat surfaces and features (referred to as "Inner-Shape Details" below). As noted by Fan *et al.* (29), these descriptions are tolerant of small changes in the angle from which the object is viewed. Though this is a useful quality for object recognition and classification, it could be problematic in pose estimation.

2.2.2.1 Overall Shape. As used in this subsection, "Overall Shape" refers to the shape of the entire object as viewed in two-dimensions. Measurements based on this shape can be broken down into two categories. The first category uses outline features. These features are calculated from treating the outline as a single continuous entity (as opposed to a collection of line segments which is used in the next subsection). The second category uses silhouette features. These features include the area enclosed by the outline of an object.

In both the outline and silhouette features, the object is represented as a binary image (no gray-scale intensities are indicated—the image is black and white with the object being represented by an area of black within an overall white background).

Outline (or Boundary Edge). As stated by Ballard (7), "In an image, the pertinent information about an object is very often contained in the shape of its boundary." In fact, early Automatic Target Recognizers (ATRs) and Automatic Target Cuers (ATCs) looked for closed surfaces by following detected edge boundaries (34). These outlines were then processed for recognition or cueing purposes.

Lucero *et al.* used outline features with moderate success (71%) to classify objects in Forward-Looking Infrared (FLIR) imagery (53). For that research, Lucero used the outline calculated from the ideal silhouette of an object which was viewed at a distance of 1 km or so. Dudani (28) also successfully used outline moments to identify objects (by using invariant moments such as Zernike moments).

Politopoulos (63) used outline moments to relate image objects to objects contained in a database. In the cited reference, Politopoulos establishes the mathematical relationship between silhouette and outline moments. Though a silhouette moment is found to be a "linear combination of products of [specific] boundary moments," the mathematical relationships still justify the use of outline moments in addition to silhouette moments.

Fourier descriptors have also been used to identify three-dimensional objects by their two-dimensional outlines (5, 19, 66). As stated by Arbter (5), "a closed curve may be represented by a periodic function of a continuous parameter, or alternatively, by a set of Fourier coefficients of this [periodic] function." Arbter normalized the curved parameterized line so as to obtain "affine" invariant Fourier descriptors (i.e., the descriptors are invariant to those out-of-plane rotations which produce a shearing effect on the object's image).

Silhouette. As opposed to the outline features of the previous subsection, silhouette features are those for which the area within the outline of an object is also used (53, 67, 85).

Though there may be some debate as to whether silhouettes may be used to classify a target (53), their use for object recognition has been moderately successful (28, 80, 3, 21, 22).

In addition to using outline moments, Dudani (28) also used silhouette moments for object recognition. However, as noted by Politopoulos (63) and referred to in the previous section, silhouette moments are linear combinations of various products of outline moments. According to Politopoulos, the specific linear combinations and products can be used to justify using both silhouette and outline moments as features.

Teague (80) used standard statistical and Zernike moments to completely reconstruct an image silhouette using only the first 16 or 18 moments. However, the shape of the object was recognizable using only the first 12 or so moments. Furthermore, Abu-Mostafa (3) showed that you need the higher-order moments when you have noise in a scene.

Instead of using standard statistical moments, Cyganski *et al.* (21, 22) used tensor representations and tensor moments. Tensor moments are calculated in similar fashion to standard moments (78); however, instead of using absolute coordinate positions, vectors to those positions are used. As stated by the researchers, this added attribute allows for standardization of the object's orientation.

In addition to the moment-based processing, global features relating to the structure of the object (and, hence, its silhouette) are also used (8, 34). These include height and width—or vertical and horizontal size—as well as the area and so forth (8). Additionally, the two-dimensional aspect ratio (34) is calculated by taking the ratio of the height to the width.

2.2.2.2 Inner-Shape Details. In this subsection, "Inner-Shape" refers to those regions or areas within the boundary of the object's image. Unlike the overall shape features which are typically binarized (black and white), inner-shape features typically rely on there being subtle or distinct changes (gray-scales) in the intensity of the inner regions. However, for IR imagery, Politopoulos (63) notes that intensity-based features are "not deemed appropriate" due to "immense variety and inconsistency presented by intensity distributions."

Furthermore, relative to IR models, Sadjadi (73) mentions that "certain relationships that form foundations of target, atmosphere, and background IR modelling are not deterministic." And, lastly, as stated earlier, Oshima (60) contends that the intensity of an area does not necessarily relate to some geometric quality of the area (though his statement was made in regard to visible imagery, it is extrapolated here to include IR imagery).

For these inner-shape features, there are three typical categories: intensity-based statistical features; sub-regions and their attributes; and textures and their qualities.

Intensity-Based Statistical Features. These features use both absolute and relative intensity values (34). The statistical features used include standard deviation, variance, average intensity, and maximum/minimum intensities.

Sub-Regions. These features include relative locations of continuous regions (73), geometric attributes such as rough geometric shapes (e.g., triangles, quadrangles, etc.) (14), and contrast between neighboring regions (9). Gilmore (34) also states that ATRs of today use image sub-regions for segmentation purposes.

Textures. In addition to using regions for target recognition, ATRs of today also use the textures within and about the regions (34). Two functions which essentially extract information about the texture within an object are the two-dimensional Gabor and Fourier transforms.

The two-dimensional Gabor transform is composed of a two-dimensional sinusoidal function (e.g., a sine or cosine function) multiplied by a two-dimensional "windowing" Gaussian envelope (6, 23, 24). The Gabor transform essentially picks out textures which correspond to the number of sinusoidal cycles at the specific orientation and extent of the sinusoidal function (23, 24).

Daugman showed that two-dimensional Gabors could be used to compress the amount of information contained within a scene (23). He later showed that neural networks could be used to determine the optimum Gabor coefficients to completely reconstruct an image (24).

As shown by Ayer (6), based on the type of sinusoid (cosine or sine), it is possible to use the Gabor for textural determination as well as for edge detection.

Whereas the Gabor transform used a Gaussian envelope to "window" the extent of the sinusoidal function, the Fourier transform does not apply such a restriction. More information on the nature of the two-dimensional Fourier transform can be found in (32, 36). For information on the Fourier transform as a feature extraction technique, see the references in the bibliography (85, 66, 50).

2.2.3 Line Measurements. As noted by Fan *et al.* (29) in the introductory section to this chapter, measurements based on lines within an image tend to be more dependent on the viewing angle than the two-dimensional surfaces from which these one-dimensional features are derived. However, as will be shown in the subsection on Inner and Outer Edges, there are certain groups of linear features which are relative invariant to the viewing angle. Line-based features include discernible edges and vertices (locations where edges come together). These two categories will be discussed in more detail in the subsections following this section.

To use these line-based features, an edge detection operation is typically performed on the two-dimensional image (75). However, as noted by Stockman (76), these operations typically produce lines which are shorter than they should be. Also, as noted by Pavlidis (61), the thickness of a real image's detected lines versus the thickness of the lines of a template (or model) can create errors when matching the template to the real image. As such, it is necessary to account for these inherent inaccuracies. Operations include using only the low-frequency spatial data (to blur the lines) (50) to more exotic means such as using fuzzy logic (61).

Stockman (76) also notes that edge detectors tend to misrepresent vertices by overshooting the "true termination of an edge in a corner." To address this problem, Stockman allows for a certain degree of "sloppiness" in the use of both edge and vertex features.

Many researchers account for the fact that lines may be obscured in an image. For example, in matching models to real objects for estimating an object's pose, Ray (65) as-

signs a label to each line to indicate whether the line is not visible, matches a line in the model/image, or is unknown. Chatravarty (17) uses "vantage-point domains" to indicate the viewing angles from which a line in a model should be visible.

Also, many researchers account for the directional nature of lines by using vector-type representations (76, 8). Stockman specifically converts line fragments into vectors (76); whereas Bart (8) represents each line with a "4-tuple" indicating the location of the centroid of the line (x and y coordinates), the length of the line, and its orientation.

2.2.3.1 Edges. In similar fashion to the earlier section on surfaces, this section is divided into measurements based on outer edge features and a combination of inner/outer edge features. Indicative of the usefulness of these features are the many different research efforts using line-based measurements (7, 8, 13, 15, 17, 34, 39, 54, 57, 61, 63, 65).

Outer Edges. As noted in the section on outline shapes, this subsection deals more with the treatment of an outline as a composition of individual line segments rather than a single continuous entity. Gilmore (34) mentions that ATRs of today use the segmented quality of an outline and the relationship among the segments to determine edge busyness. This feature is then used for target recognition. Also, Ray (65) uses this segmented nature of the outline for pose estimation.

Inner/Outer Edges. This subsection refers to all lines detected in an object's image, not just those associated with the outline of the object. After being detected as stated in the introductory paragraphs to this section, these lines are then used according to: each line's geometric parameters; the geometric parameters for groups of two or more lines; and, mathematical manipulations of the lines without necessarily distinguishing any one line or line group from another (e.g., using Hough transforms).

As examples of using just individual lines, Bidlack's work and Ballard's work show the types of features extracted (7, 11). Bidlack (11) calculates line segments based on the end points of the line. The features measured (referred to as geometric data) are pose dependent and include magnitude, length, relative position, and orientation. Ballard (7) uses vector

quantities to represent the points on a line. Instead of each point representing a gray-scale value of intensity, Ballard uses a gradient representation for each point on the line. This representation uses a directional quantity to indicate the direction of grav-scale change, and a magnitude quantity to indicate the severity of the gray-scale change.

Instead of treating each line as a individual entity, Brooks (13) and Lowe (54) use groups of line segments which are invariant for small changes in the viewing angle. These line-group geometric parameters include connectivity, collinearity, parallelism, proximity, and symmetry. These linear groupings are then used for feature comparison.

Without distinguishing among lines as is done with the geometric parameterizations mentioned above, the Hough transform is a clustering technique (37) which accumulates points based on the transformations from lines of a model to lines of an image (7). Details on this widely used mathematical treatment of linear features are provided below.

For each linear feature of a model, there is a specific set of translations and rotations which will bring the feature into correspondence with a linear feature in an image. If a coordinate system is used which has separate axes for each of the translations and each of the rotations, a transformation (the set of translations and rotations) would be represented as a point in this coordinate system's space (referred to as accumulator space) (7). A range of transformations would be represented as a "hypercloud" (a nebulous region of points in the accumulator space).

Consequently, for each linear image feature, there would be a point (or region) in accumulator space indicative of the transformation (or range of transformations) needed to bring that linear image feature into correspondence with a linear model feature (37). In theory, then, the largest cluster of points in accumulator space would represent the most probable transformation operation between the image and the model (7).

However, as noted by Grimson (37), since the Hough transform computes "all the transformations from a model to an image by pairing each model feature with every possible image feature," the number of large false clusters could potentially be high. Grimson ob-

serves, "there will always be false matches if there are more than 47 features in the image." (Grimson notes that 500 linear features is not an unreasonable number of linear features to find in a real image.)

The exact number of false matches can be exceedingly large when considering six degrees of freedom between real images and models. Grimson states that, using 5-pixel quantized translations and 5-degree quantized rotations, there are 10^{12} quantized transformations which can be represented. As such, many researchers use filtering concepts to restrict the accumulator space to a more reasonably sized space and to improve accuracy of the technique (37, 7, 12, 47, 75, 81).

2.2.3.2 Vertices. Vertices occur at intersections of line segments. Research with vertices essentially concentrates on the relative locations of the vertices in an image either with or without regard to the shape of the vertices in the image. Representative examples are presented below.

Location of Vertices. Research with this feature focuses on just the relative locations of the corners in a image without regard for the shape of the vertices. Then, either the relative locations are processed, or the vector representations are processed (9, 11, 17, 18).

Vertex Shape. In addition to determining the relative location of the vertices within an image, this category of linear-feature extraction also takes into consideration the shape of the vertices (13, 76, 81). Once specific shapes have been located, either the relative locations of a specific shape is processed (81) or vectors are established between specific shapes according to some rule (76).

Teh and Chin (81) find the relative locations of either L-shaped or U-shaped vertices (which are more ⊔ shaped than indicated in the letter "U"). The location of the vertex is represented by a point at the centroid of the base line-segment (i.e., the horizontal portion of the L or U shape). As noted by the researchers, this method is a "variant" of the Hough transform. However, this method does not really account for the directional nature of the

lines (i.e., their orientation). As noted by Ballard (7), the original Hough transform work did not account for line orientation and was "inferior" to later work that did.

Stockman (76) finds the X-, Y-, T-, and L-shaped vertices and uses a rule to establish vectors between specific vertex pairs. For example, a vector may be established from an X-shaped vertex to an L-shaped vertex, but not vice versa. Additionally, the rule may exclude vectors between some pairs (e.g., no vectors between X- and T-shaped vertices). The effect of the rule is to limit the varieties and number of vectors which have to be processed. As noted by Stockman in (76), the directional nature of these vectors is required to completely estimate the pose of an object within an image.

2.2.4 Point Measurements. This last (and, as noted by many researchers, most primitive) feature is often the final step in many of the earlier feature processing routines. However, as will be shown below, some feature extractions start with point measurements (such as the use of standard landmark features). Though, for the most part, less primitive features such as regions, lines and vertices ultimately are processed according to some point which represents the location of the feature (1, 11, 18, 30, 39, 54, 81, 82, 86). Point measurements include those points which correspond between a model and an object's image, between multiple views of a object, or between different portions of the object itself.

Once the relative locations of the points are known, some mathematical manipulation is performed to estimate the pose of the shape from which the points were extracted (or, in the terminology of many researchers, the points are used to determine camera location—to use an old cliche, it's "six of one and a half-dozen of another"). If the points are those which correspond between a model and an object's image, the model can be used to determine the pose of the object (41, 54, 77). Additionally, if the points are those which correspond between multiple views of an object, the correspondence may be used to determine the pose of the object (82).

2.2.4.1 Standard Landmark Features. These features are typically items such as "Hot Spots" in an IR image (53) or holes in a visible image (39). The locations of these

features are related to the locations of features determined by corners, edges, and so forth (17, 18, 30, 39, 67).

2.2.4.2 Three-Dimensional Point Projections. As stated above, once the relative locations of points are determined, some processing of the point groups is used to estimate the pose of the shape from which the point groups were obtained. In this subsection, four types of point groups are discussed: point pairs; three or more points in perspective images: four-point planar; and, four-point non-planar. It should be noted that perspective imagery reduces to isometric imagery if the range is large (so that variations in the dimensions over the extent of a target are imperceptible).

It should also be noted that, though it seems to consider the object to be transparent so that you see hidden corners, processing these points as though originating from a volume in three dimensions is still a candidate technique for the purposes of this research. For example, the relative locations of the tip of a barrel, hot-spot centroid, and external corners of a tank represent a volume which may be completely visible in real imagery from several real-world vantage points.

Point Pairs. These are two points detected within an image (76), or determined to be corresponding points between two views of an object (82), or determined to be corresponding points between a two-dimensional projection of a three-dimensional model and a two-dimensional image (41, 54, 77), etc. If the two points are detected within a single image, they are typically represented by a vector between the two points (76) or by their relative locations within the image (76, 51). In all cases, the point pairs are considered collectively for purposes of object pose estimation.

Perspective-n-Point. This category includes three-point (P3P), four-point (P4P) and six-point (P6P) data sets from the perspective image of an object:

- P3P (11, 30, 54). Without restrictions applied, three points can result in four possible orientations when viewed in three-dimensions (30). However, this number can be reduced by considering the visibility of the features (54) or by considering the stability

of the object for which some poses may not be feasible (such as a tank resting on its barrel only) (39).

- P4P (30). Four points are shown to have two possible orientations when viewed in three-dimensions (30). But, as noted above, restrictions may be applied to eliminate some of the ambiguities.
- P6P (30). Six points are shown to have one unique solution (provided the points do not degrade to P3P or P4P).

Four-Point Planar. The constraint of planarity for four points seems reasonable given objects composed of quadrilateral facets; however, as noted by Bidlack (11), feature extractors may not be able to find four planar points (even if they existed). This limitation suggests using a more general four-point method such as the P4P or the tetrahedra-volume method.

Tetrahedra Volume (Four Non-Planar Points). Haralick (40) establishes a closed-form solution for pose estimation of the four-point problem. And though the work of Fischer and Bolles (30) might suggest otherwise, Abidi and Chandra (1, 2) establish conditions for four points (noncoplanar and noncollinear) for which they contend there is one unique solution to the pose estimation problem. In (2), the researchers convert the four located points into their corresponding six linear components (four edges and two diagonals) and process the points based on their linear characteristics. From this processing, they contend one can fully determine the orientation of the tetrahedron.

2.3 Summary

In summary, the types of features and feature operations listed in this chapter were segregated according to the dimensionality of the measurement:

- Volume Features. Used range data, shading, and multiple views to extend a two-dimensional image to a three-dimensional representation of an object. However, these features as not truly volumetric in that they are based only on visible sides of an

object. As for volumetric techniques, the extended Gaussian image was identified as a candidate because it can be created using three-dimensional computer models.

- Surface Features. Treated the object as a single flat surface and used overall shape characteristics (outlines and silhouettes) or treated the object as a collection of individual flat surfaces and used inner-shape details (intensity-based statistical features, sub-regions, textures).
- Line Features. Use lines detected in an image. Based processing on edges and vertices (junctions of edges). For the vertices, both their locations and their shapes were used in pose estimations.
- Point Features. Specifically selected from the image (e.g., standard landmark features) or derived from the other "less primitive" features. Used to estimate the pose of an object via three-dimensional point projections. By considering the types of points (point pairs, Perspective-n-Points, four-point planar, four-point non-planar), equations in closed form are available to predict the three-dimensional pose of an object; however, some ambiguities may exist which can be eliminated given vantage-points and visibility/stability considerations.

III. Methodology

3.1 Introduction

This chapter begins with a justification of the various methods selected for this research. The use of shape moments as features for pose estimation of the synthetic imagery is supported by discussing the pros and cons of the various methods presented in Chapter 2. The rationale for sorting the data into 15-degree angle bins is given along with the rationale for the hierarchical sorting technique (i.e., sorting the data into consecutively smaller angle-bins). And, finally, within the justification section, rationale is provided for just using one parameter (sensor viewing-angle) to determine pose of an object.

Following the justification section are the research details. These include details on image generation and manipulation, feature calculations, neural networks software/configurations, and the method for using the neural-network software to sort the data into the angle-bins.

3.2 Justification of Methods Used.

This section provides rationale for using the various methods within this research effort. It includes discussions on the use of shape moments as features, the use of 15-degree angle-bins as the "resolution" of the pose estimation, the use of the technique of sorting the imagery into consecutively smaller angle-bins, and the use of the sensor viewing-angle as the critical parameter for pose estimation.

3.2.1 Use of Shape Moments for Features. As shown in Figure 1, for this research, shape moments were calculated from the binary silhouette and outline images of the object. Based on the review of the many methods used for recognizing three-dimensional objects from their corresponding two-dimensional IR imagery, shape moments were selected as the best starting features. Shape features have been shown to be useful in recognizing and determining the pose of an object (5, 7, 21, 22, 34, 53, 66, 67, 84, 85). Furthermore, the specific use of shape moments (3, 28, 63, 71, 80) has been shown to be moderately successful.

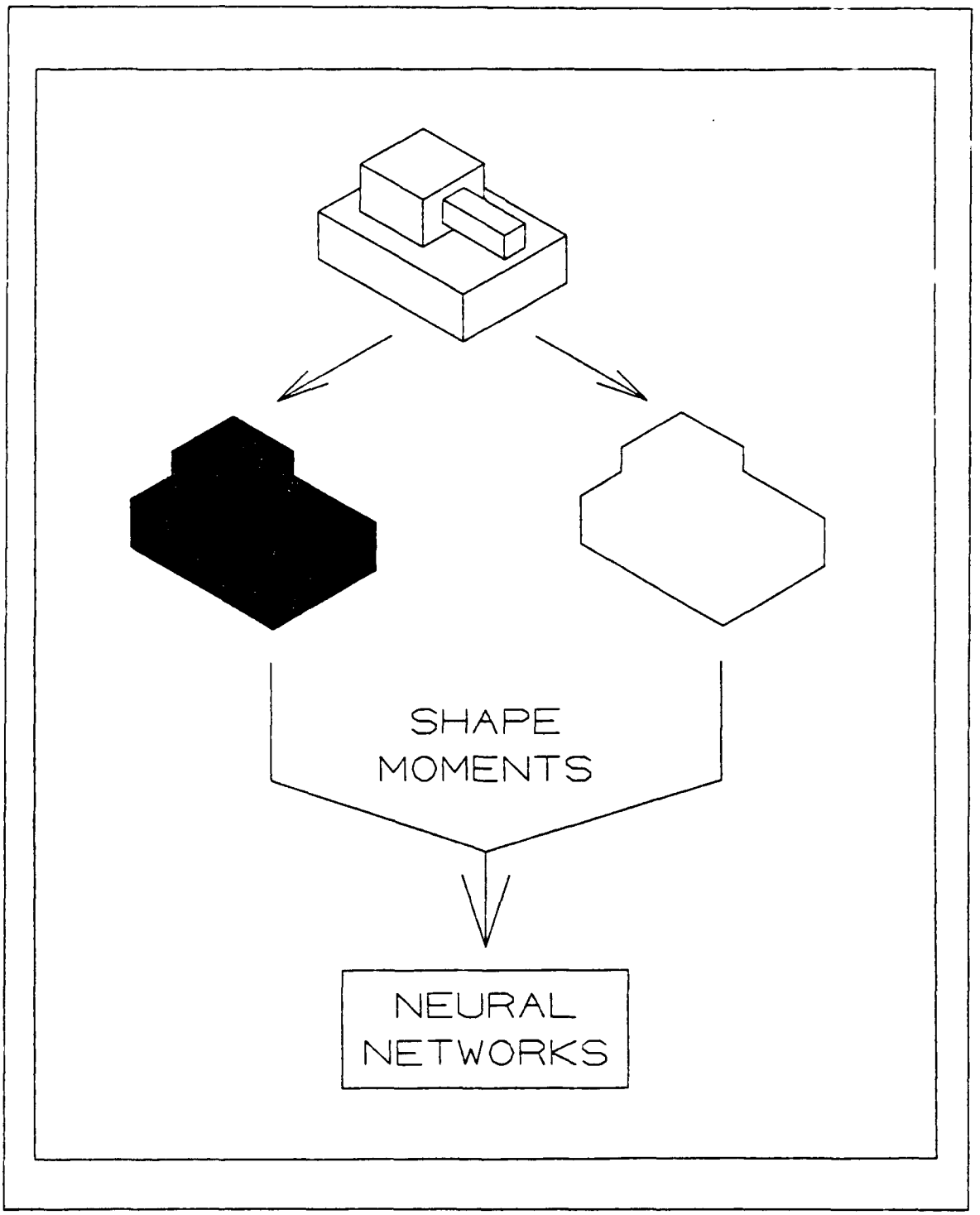


Figure 1. Flowchart showing information processing for this research. Shape moments are calculated from the silhouette and outline images. The data are then processed using neural networks.

Also, this feature appears to be most extendable from the synthetic imagery to real-world imagery—the only true accuracy required is that of the geometric representation of the model. However, it should be noted that there are sensor views for which the synthetic imagery does not appear to mimic real-world imagery well enough for mathematically stringent shape calculations (see Chapter 4).

As a general note, intensity-based features were not deemed good first choices due to perceived problems with the modeling software and reported problems for IR intensity-features as a basis for object recognition (59, 63, 73). Intuitively, model-based methods using the IR intensity distributions would at least require

- good correlation between the geometric parameters of the model and the item being modeled,
- good correlation between the relative temperatures of the surfaces of the model and item being modeled,
- good modeling of the conductive/radiative properties of the materials composing the item being modeled.

As will be seen in Chapter 4, the model did not appear to adequately account for either the geometric or temperature requirements to the extent that the intensity-based methods were selected as the first choice for this research. Additionally, as noted in Chapter 2, Sadjadi (73) states that the IR relationships needed for modeling are not fully deterministic. Politopoulos (63) notes that the IR intensity representations vary greatly and can be inconsistent due to the distributions within the target. Oshima (59) points out that areas of equal perceived intensity do not necessarily represent geometric areas of an object—a further requirement that very good correlation is needed between the synthetic and real-world imagery (so as to eliminate the need for associating intensity with specific geometric attributes).

For the four categories (and some of their sub-categories), an itemized rationale is provided below. This rationale is based on the applicability of the feature or technique to

the specific problem of pose estimation of three-dimensional objects based on their two-dimensional IR imagery as generated by a computer modeling program.

1 Volume Features.

(a) Using Range Data. Though pixel-registered range and IR model-generators are available (52), this added dimension was not immediately applicable to the modeling/rendering software at hand.

(b) Using Shape from Shading. This technique is based on intensity within an object. As noted above, techniques of this type were not selected as a good first choice.

(c) Using Multiple Views. This option is still a candidate option. The ability to create imagery of objects at any orientation is a strong feature of the modeling software. However, in using this method, the silhouettes and outlines of the objects would be used (to avoid the intensity problems noted earlier). Therefore, this method is deemed a second choice to the silhouette and outline techniques. However, this technique does require multiple-views for real-world imagery and, in that regard, is somewhat outside the initial intent of this research.

(d) Using the Extended Gaussian Image (EGI). As with the range data, this technique is not immediately extendable to the modeling software. However, the modeling software could be used to generate the EGI which may have other properties useful for excluding certain poses once an initial routine has guessed the pose of an object.

2. Surface Features. Since the software models the object as a collection of facets at specific temperatures (4, 72), techniques which use shape are attractive. The only true requirement is accuracy between the geometric qualities of the model and the object modeled.

(a) Overall Shape (Outline, Silhouette). As noted above, these features were determined to be good first-choice features. Furthermore, these features (often referred

to as "scalar" features) are sometimes used to narrow down the options for further feature processing (inner-shape details, essentially) (8). As stated by Bart *et al.* (8), "scalar features are considered to be so important, that objects that differ too much in them will certainly not match."

(b) Inner-Shape Details. These features tend to rely on the intensity distributions of an object. As noted above, these types of features were not deemed good first choices.

3. Line Features. As with the surface features, these features appear to be tied to the facet-like nature of a model. However, these features presume either accurate correlation between the model and the real-world object, or they presume the intensity distribution of real-world objects represent geometric qualities of the object. As noted above, these presumptions are the reason these features were not selected as first-choice features.

4. Point Features. These features were deemed good second-choice features since all that is required is that there are a few points which can be correlated between an object in the synthetic imagery and the object in the real-world imagery. The selection of a standard landmark feature (such as the centroid of the hottest region) could be used in conjunction with shape related features to fully confirm the pose of an object (especially since the hottest spots would seem to be associated with engine compartments). However, it should be noted that a hot barrel and hot treads (for a tank) could shift the centroid of the hot spot away from the engine compartment. As such, this feature was not selected as an optimum first-choice feature.

   It should also be noted that, in using point features, the vantage-point domains should be determined to ensure visibility of all the required points from viewing angles likely to be encountered in real-world imagery (17). These vantage-point domains could also be used to describe the limitations of specific point features and techniques.

3.2.2 Use of 15-degree Angle-Bins. Based on research by Le (49), a 15-degree rotation of the object in the plane of the base of the object (which corresponds to the viewing plane only for top or bottom views of the object) was selected as the resolution for this research. Le based his choice on the work by Hubel and Wiesel (45) and Gizzi *et al.* (35).

This technique (though not referred to specifically as using "angle-bins") has been used by other researchers. Korn (47) mentions calculating adjacent views of a model and grouping those views that are "feature equivalent." Then, a representative view is used for the entire class of feature-equivalent views. According to Fan *et al.* (29), Ikeuchi (46) generated object models under various viewer directions and grouped the apparent shapes. In these types of representations, Bart (8) notes that it's important to "detect gaps in the series of orientations and to delete superfluous orientations."

3.2.3 Use of Hierarchical Sorting. This decision was based on the nature of the shape moments selected for features. Moments indicate the distribution of an object's mass about some reference entity (a point, a line, etc.) (43). As noted by Teague (80), "...if only moments up through second order are considered, the original image is completely equivalent to a constant irradiance ellipse having definite size, orientation, and eccentricity and centered at the image centroid." For example, second order moments relate a two-dimensional object to an ellipse with the same principal axis of the object (the axis of least inertia) with the same mass distribution about the principal axis (43).

Based on this analogy to an ellipse, it can be noted that ellipses tend to exhibit symmetries which could interfere with an unambiguous pose estimation. Consequently, it was judged that there would most likely be ambiguous sets of data if the entire 360 degrees of rotational freedom being measured were used for sorting into the 15-degree angle-bins.

As such, a method of hierarchical sorting was decided upon. In this method, the data is used to sort the object into consecutively smaller angle-bins. For example, in one possible sorting scheme, the object would be sorted into one of the two 180-degree angle-bins constituting the entire 360 degrees of possible rotations. Further processing would sort the

object into consecutively smaller angle-bins: 180-degree angle-bin to 90-degree angle-bin; 90-degree angle-bin to 45-degree angle-bin; and, finally, 45-degree angle-bin to 15-degree angle-bin. Various permutations of these "paths" were explored to find the optimum path.

3.2.4 Use of Sensor Viewing-Angle as Critical Parameter. The sensor view-angle (see Figure 3) determines the rotation of the object within the plane of the base of the object. Since it is possible to use LIDAR or other techniques to determine the orientation of the ground around a target, the base-plane rotation of the target was deemed the most critical parameter. However, once a set of features had been selected which could be used to determine the base-plane rotation of an object, the effects of varying the sensor depression-angle were explored.

3.3 Research Methodology.

In this section, details are provided on the synthetic imagery software, on the manipulation of the data in order to extract the features, on the mathematics of the features, and on the various conventions used in this research for configuring and using the neural network software. The following list gives the order in which these will be discussed:

- Creating the images with the software (Image Generation).
- Creating the outlines and silhouettes from which the features will be extracted (Image Manipulations).
- The mathematics of the features initially extracted (Feature Calculations).
- The processing of these features once extracted (Neural Networks).
- The method of sorting the data into consecutively smaller-sized angle-bins (Sorting into Angle Bins).

3.3.1 Image Generation. The synthetic imagery for this research was generated using the Georgia Tech Signature (GTSIG) prediction software and their Scene Generation (SCN-GEN) rendering software. This software combination was designed to produce research-grade

infrared imagery of various models (26). For this research, the model used was a T62 main battle tank sitting stationary in the afternoon sun with its engine running (the engine has been running for six hours continuously). Also, in the simulation, the tank is situated at Eglin AFB in Florida during late-February (the location and date are specified, as mentioned below, in the **rg_t62_idle.in** file).

To generate an image, the program GTSIG is first used to produce a radiance map for the model. Then, SCNGEN is used to combine this radiance map with the model's geometric (or facet) file to generate imagery of the model in whatever orientation and scale the user specifies.

The **rg_t62_idle.in** file defines parameters such as geographical location, day of the year, and so forth. This file also determines the hour of the day for which the simulation is to start (such as 0800 in the morning). When generating imagery, a time of day is entered and the imagery supposedly reflects the position of the sun in the sky for that geographical location. Additionally, the time-of-day parameter for image generation supposedly determines how much heat has been conducted to various parts of the model; however, as can be see in the synthetic imagery (e.g., Figure 7), the heat from the engine compartments does not appear to have spread very far after the engine has been running continuously for six hours.

3.3.1.1 Image Orientation. As shown in Figures 2 and 3, there are five principle parameters which refer to the orientation of the rendered image (the effects of changing these parameters are shown in Figures 4 and 5—note that, as shown in Figure 5, changing the target parameters can produce anomalous imagery):

- Target azimuth-angle. This angle determines the rotation of the object relative to the compass settings (North, South, etc). This angle is used to place the object relative to the sun's position. Changing this angle does not change the orientation of the object relative to the sensor, though (since sensor position is relative to the front of the target and not the target's absolute compass heading).

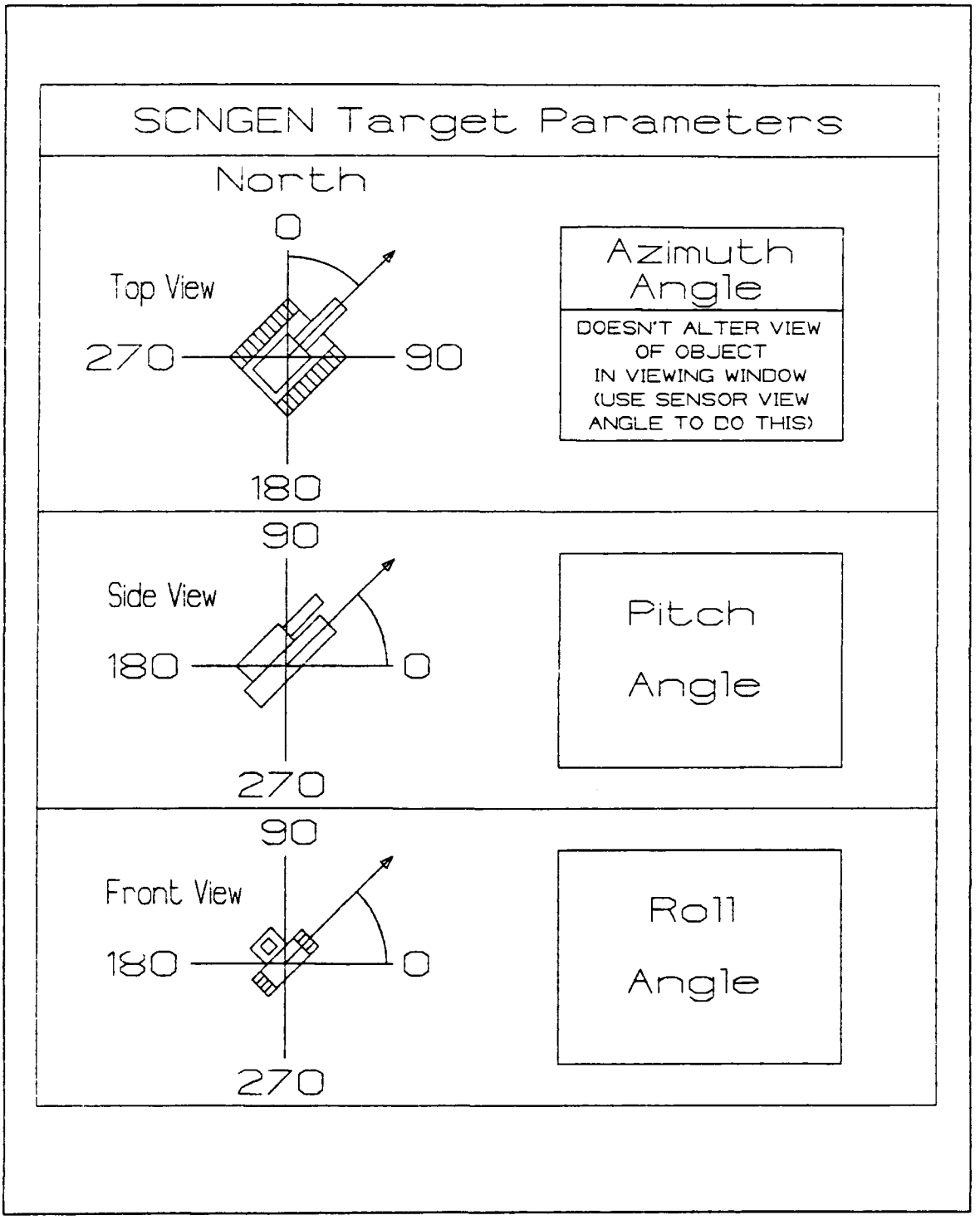


Figure 2. SCNGEN target-relative parameters. These angles determine the orientation of the target relative to the sun or to the sensor (see text).

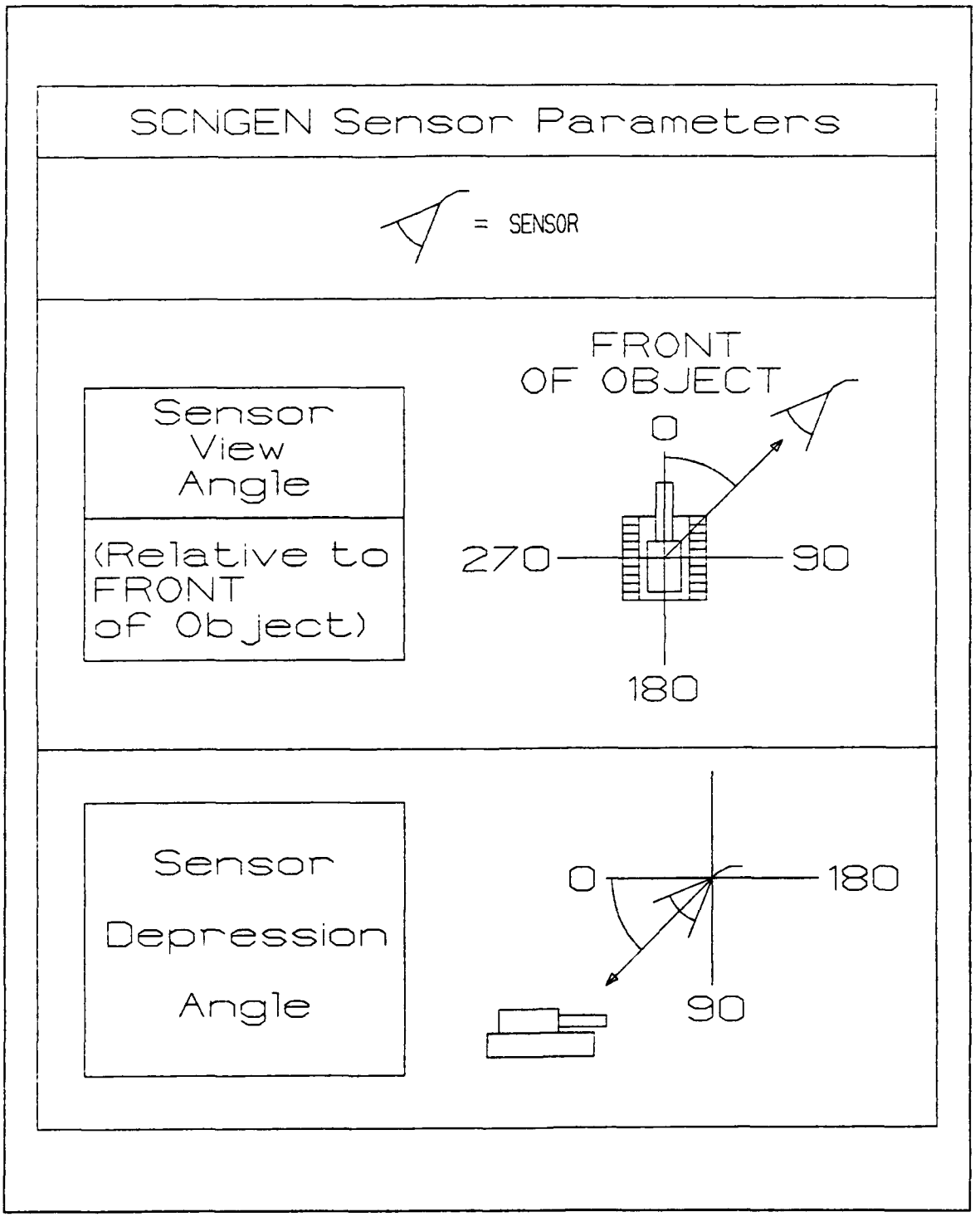


Figure 3. SCNGEN sensor-relative parameters. These angles determine the orientation of the sensor relative to the target.

Note, however, that though the orientation of the object does not change with changes in target azimuth-angle, the field of view did change (even though it was not supposed to change). This apparent flaw in the software appeared to occur only at target azimuth angles which were integer multiples of 45 degrees.

- Target pitch-angle. As with an airplane, this angle determines the pitch of the target. Note in Figure 5 that the object is no longer filling the field of view as specified in the image generation program (it should fill 75% of the field of view). This is another apparent flaw in the program.
- Target roll-angle. As with an airplane, too, this angle determines the roll of the target. Again, note in Figure 5 that anomalies occur when the object is rolled 90 degrees—the object is not centered in the field of view, but is truncated at the edges of the imagery. This, too, appears to be a flaw in the program.
- Sensor view-angle. This angle, relative to the front of the object, determines the angle from which the target is viewed (to create a front *view*, *side view*, *or oblique-angle* view and so forth).
- Sensor depression-angle. This angle determines whether the target is viewed from ground-level, low-altitude, high-altitude, or overhead.

Only three of these angles are required to produce an image at any orientation: **sensor view-angle**, **sensor depression-angle**, and **target roll-angle**. Of the other two parameters, changing the target azimuth-angle doesn't alter the orientation of the image that is viewed and changing the target pitch-angle is redundant to changing the sensor depression-angle unless one is accounting for the position of the target relative to the sun. Note, though, that changing the target parameters tends to produce anomalous imagery.

3.3.1.2 Image Scale. To change the scale of the image (and, hence, effectively alter its apparent range), the Field of View (FOV) is adjusted using SCNGEN. Though SCNGEN has a "range" parameter which can be varied, varying this parameter produces no

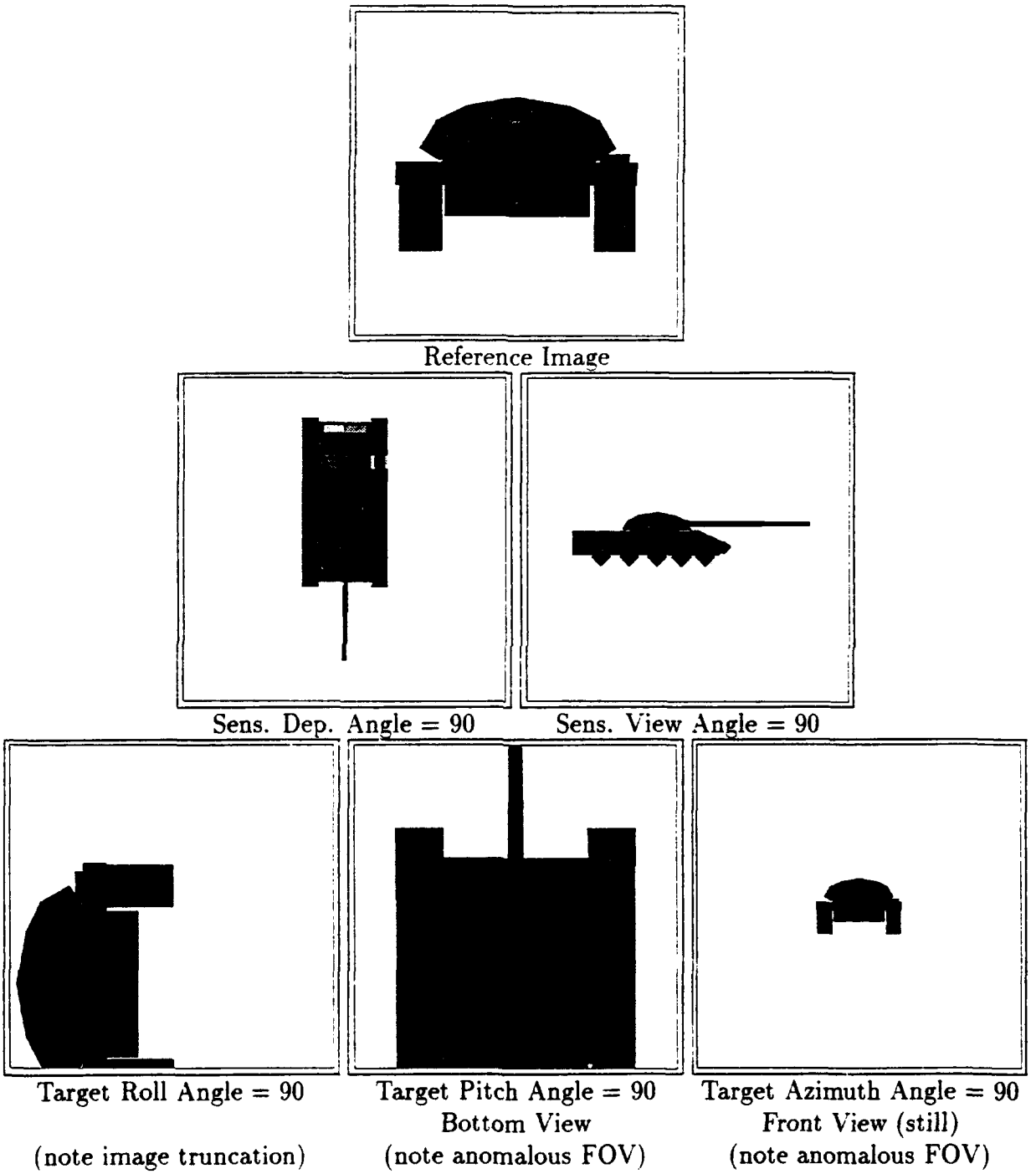


Figure 4. Effects of changing sensor and target parameters. Reference object is a T-62 tank facing the viewer (sensor/target parameters set equal to zero). NOTE: Changing the target parameters produces anomalous results. For these images, the field of view (FOV) was **not** changed.

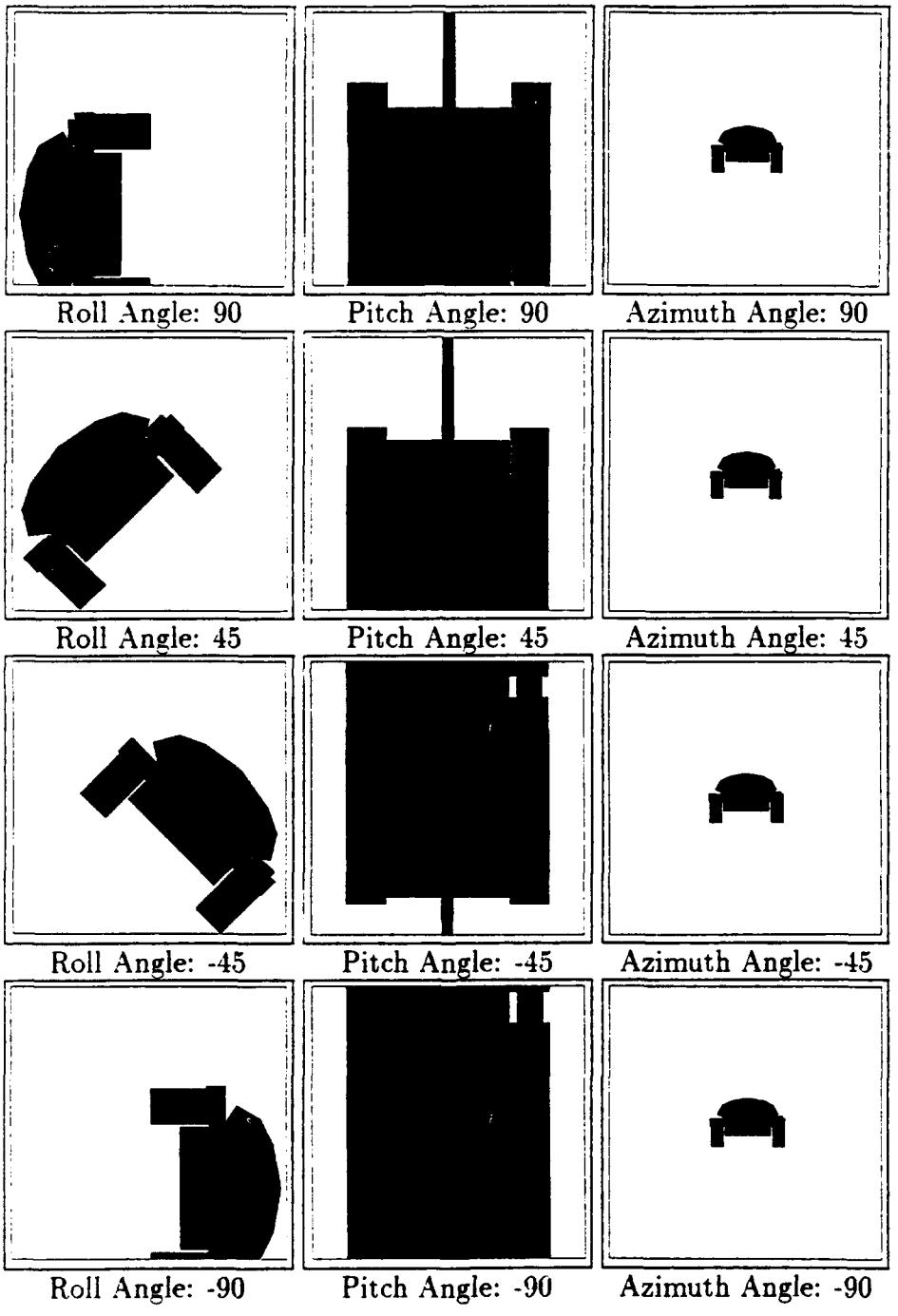
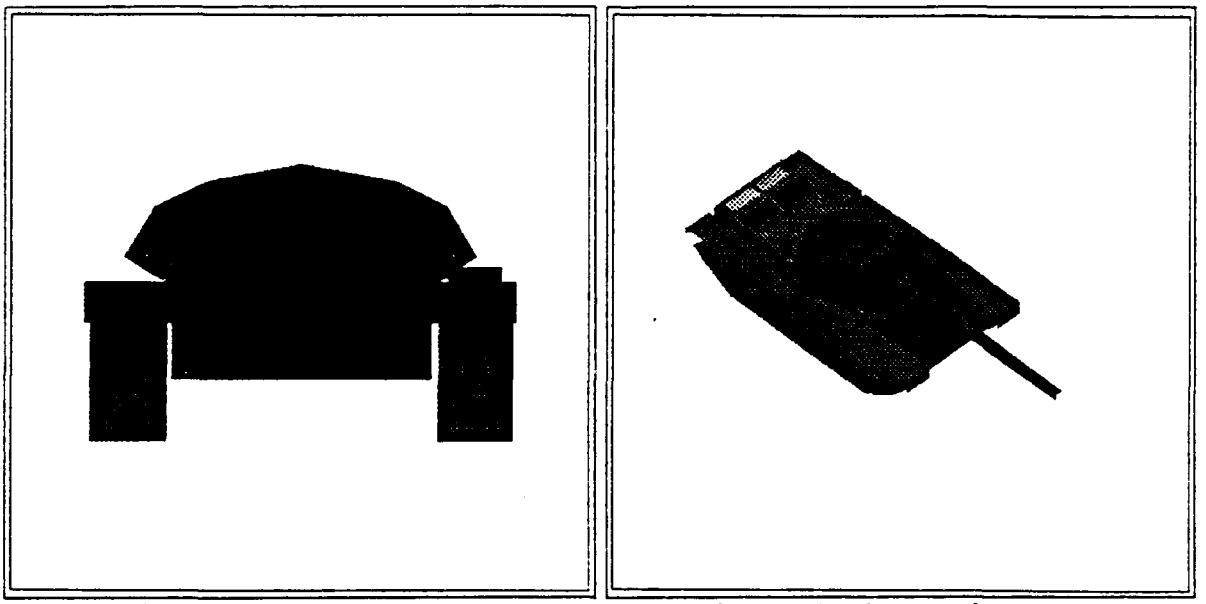


Figure 5. Image anomalies from changing the target parameters.



Sensor Angles: 0 degrees. Sensor Angles: 45 degrees.

Figure 6. SCNGEN images with FOV = 0.75 for both images, but with sensor depression-angle and sensor view-angle both equal to 0 degrees for the image on the left and both equal to 45 degrees for the image on the right.

effect on the resulting image. The documentation for the SCNGEN program (26) suggests this parameter will allow atmospheric effects to be taken into account; however, there were no differences in the imagery as this range was varied from 100 meters to 6000 meters (which represents the range over which real-life imagery is typically available).

The FOV represents the percentage of the viewing area occupied by the model along either the x-axis or the y-axis of the viewing area (e.g., FOV = 0.8 means the model is scaled until one of its dimensions fills 80% of either the horizontal or the vertical field of view). As such, it is mathematically laborious to calculate the apparent range for a given orientation and FOV. Additionally, for a given FOV, different orientations will produce different apparent ranges (see Figure 6).

3.3.2 Image Manipulation. The file SCNGEN creates is a 512x512 (user-selected) floating-point image file which includes a header describing the parameters used in generat-

ing the image. This file was converted to a gray-scale image (256 gray levels) by the program "img2dis" which is available through the thesis advisor for this research. The ".DIS" extension was chosen so as to be consistent with the extension assigned by RDIRMA (another Georgia Tech program which, among other things, can convert the floating-point image file to a gray-scale image file). The "img2dis" program was written, however, because of problems with automated use of the RDIRMA program: RDIRMA would lock up when the UNIX input-redirection command was used (therefore, the user could not automate the inputs and had to interact with RDIRMA manually for each image—for the many hundred images required, this was impractical).

Once converted to a gray-scale image, the image could be further manipulated to obtain the silhouette image and outline image of the model (see Figure 7). The program "dis2tru" produced the silhouette image. The "tru" was chosen since this image can be used to "truth" the original image (i.e., mask out the background from the original gray-scale image). The program "tru2otl" produced the outline image from the silhouette image. Both of these programs are available through the thesis advisor for this research.

3.3.3 Feature Calculations. The features used in this research were calculated as explained below. In general, the features were extracted from the binary silhouette and outline images.

3.3.3.1 Shape Moments. The "zeroth" through to the third-order shape moments were calculated as shown below. These moments are referred to as shape moments since they were calculated from the silhouette image (or binary-image: 0 outside the object, 1 within the object) and from the outline image (which is also binary) (Figure 7). These moments depend only on the shape that is viewed—they do not require inner-image intensity features such as hot-spots, cold-spots, etc.

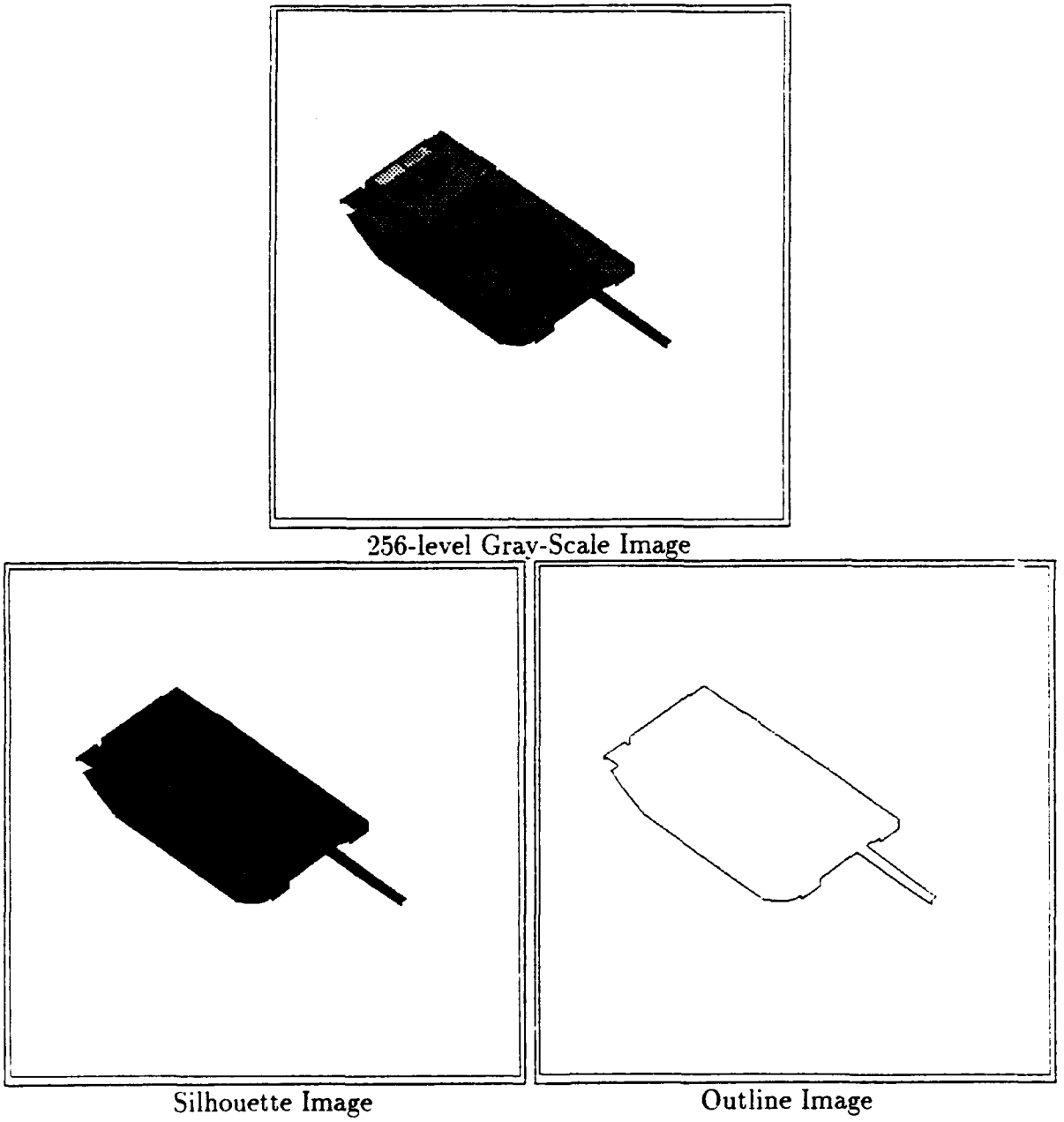


Figure 7. Corresponding gray-scale, silhouette, and outline images.

The "zeroth" and first-order moments (M) were calculated (44, 71):

$$M_{mn} = \sum_{x=0}^{511} \sum_{y=0}^{511} x^m y^n B(x,y)$$

where $B(x,y)$ is the binary-valued (0 or 1) silhouette or outline image (see Figure 7).

The "zeroth" order moment (M_{00}) gives the area of the object. This area is then divided into the first-order moments to obtain the location of the object's centroid:

$$x_c = M_{10}/M_{00}$$

$$y_c = M_{01}/M_{00}$$

These centroid-coordinate values are then used to make the remaining moments shift-invariant (44, 71):

$$M_{mn} = \sum_{x=0}^{511} \sum_{y=0}^{511} (x - x_c + 256)^m (y - y_c + 256)^n B(x,y)$$

where $m + n \geq 2$, and the 256 is for the x and y coordinates of the center of the overall image (half of 512). These adjustments have the effect of shifting the centroid of the object to the center of the overall image space.

To make the moments scale-invariant, various manipulations were investigated. The initial features calculated were:

$$M_{20}/M_{02}$$

$$M_{20}/M_{11}$$

$$M_{02}/M_{11}$$

$$M_{30}/M_{03}$$

$$M_{30}/M_{21}$$

$$M_{30}/M_{12}$$

$$M_{03}/M_{21}$$

$$M_{03}/M_{12}$$

$$M_{20}/M_{00}$$

$$M_{02}/M_{00}$$

$$M_{11}/M_{00}$$

$$M_{30}/(M_{00})^2$$

$$M_{03}/(M_{00})^2$$

$$M_{21}/(M_{00})^2$$

$$M_{12}/(M_{00})^2$$

where M_{mn} represents the silhouette moments or outline moments as appropriate.

3.3.3.2 Orientation Angle. The second-order moments were used to calculate the orientation angle, θ, of the ellipse that could be drawn around the object (80):

$$\theta = \frac{1}{2}\arctan\left(\frac{2M_{11}}{M_{20} - M_{02}}\right)$$

where M_{mn} represents the silhouette or outline moments as appropriate.

3.3.3.3 Image Aspect Ratio. Two aspect ratios were calculated. The first was overall-height to overall-width. That is,

$$\frac{y_{max} - y_{min}}{x_{max} - x_{min}}.$$

The second aspect ratio was calculated by effectively slicing the object first into individual vertical slices and finding the longest vertical slice, and, then, slicing the object into individual horizontal slices and finding the longest horizontal slice. The length of the maximum

horizontal slice was then divided into the length of the maximum vertical slice to obtain this second aspect ratio.

3.3.4 Neural Networks. Once the features were extracted from the imagery, the numerical data was analyzed using a neural network (see Figure 1. The goal of this analysis was to determine the orientation angles for the object within the imagery. Once determined, these angles could be used by a modeling program (such as the GTSIG/SCNGEN combination) to generate synthetic imagery for comparison (or for correlation).

3.3.4.1 Neural Graphics Configuration. The program **Neural Graphics** was used for all neural-network analysis. This program, developed at AFIT, can analyze data using various neural-network paradigms and configurations (79). The paradigm used for this research was "backpropagation with momentum." After experimenting with various configurations, the configuration selected for this research was: two hidden-layers—the layer closest to the input had ten nodes; the layer closest to the output had five nodes. Additionally, the starting node weights were randomly initialized.

The configuration for Neural Graphics is determined by the ASCII text file **setup.fil** (79). For this research, **setup.fil** contained the following:

```
3
10 5
random
0.0
1
data.file
1
output.data
```

Each line of the input file is explained below.

```
3             (# of layers:  3 layers = 2 hidden layers)
10 5          (# of nodes in the hidden layers)
random        (file for node weights--use random weights)
0.0           (amount of noise to be added to inputs)
1             (paradigm--BackPropagation with Momentum)
data.file     (input file containing training/test vectors)
1             (statistically normalize the inputs)
output.data   (file for output data)
```

3.3.5 Sorting into Angle-Bins. The goal of this research was to accurately determine the orientation angles of an object using neural networks. To accomplish this, the orientation of the object is determined by finding the consecutively smaller angle-bins which apply to the model (see Section 3.2.3 for rationale). An example of this technique is given below.

As shown in Fig 8, in one possible sorting sequence or "path," the image is first coarsely sorted according to the quadrant from which the target is viewed (where the quadrant corresponds to a 90-degree angle-bin). The four quadrants correspond to sensor view-angles of 0-89 degrees, 90-179 degrees, 180-269 degrees, or 270-359 degrees. Then, given the 90-degree angle-bin (or quadrant), the object is sorted into the corresponding 15-degree angle-bin. For this example, five neural networks would be required for the sorting. The first would be trained to coarsely sort the image into a 90-degree angle-bin. The other four would be trained to sort the image into a 15-degree angle-bin (one network for each of the four possible 90-degree angle-bins). As another possible sorting sequence, the image can be sorted into the correct 45-degree angle-bin given the 90-degree angle-bin, and, then, sorted into the correct 15-degree angle-bin given the 45-degree angle-bin.

3.4 Summary

This chapter provided justification for the various methods selected for this research: the use of shape moments as features for pose estimation; sorting the data into 15-degree

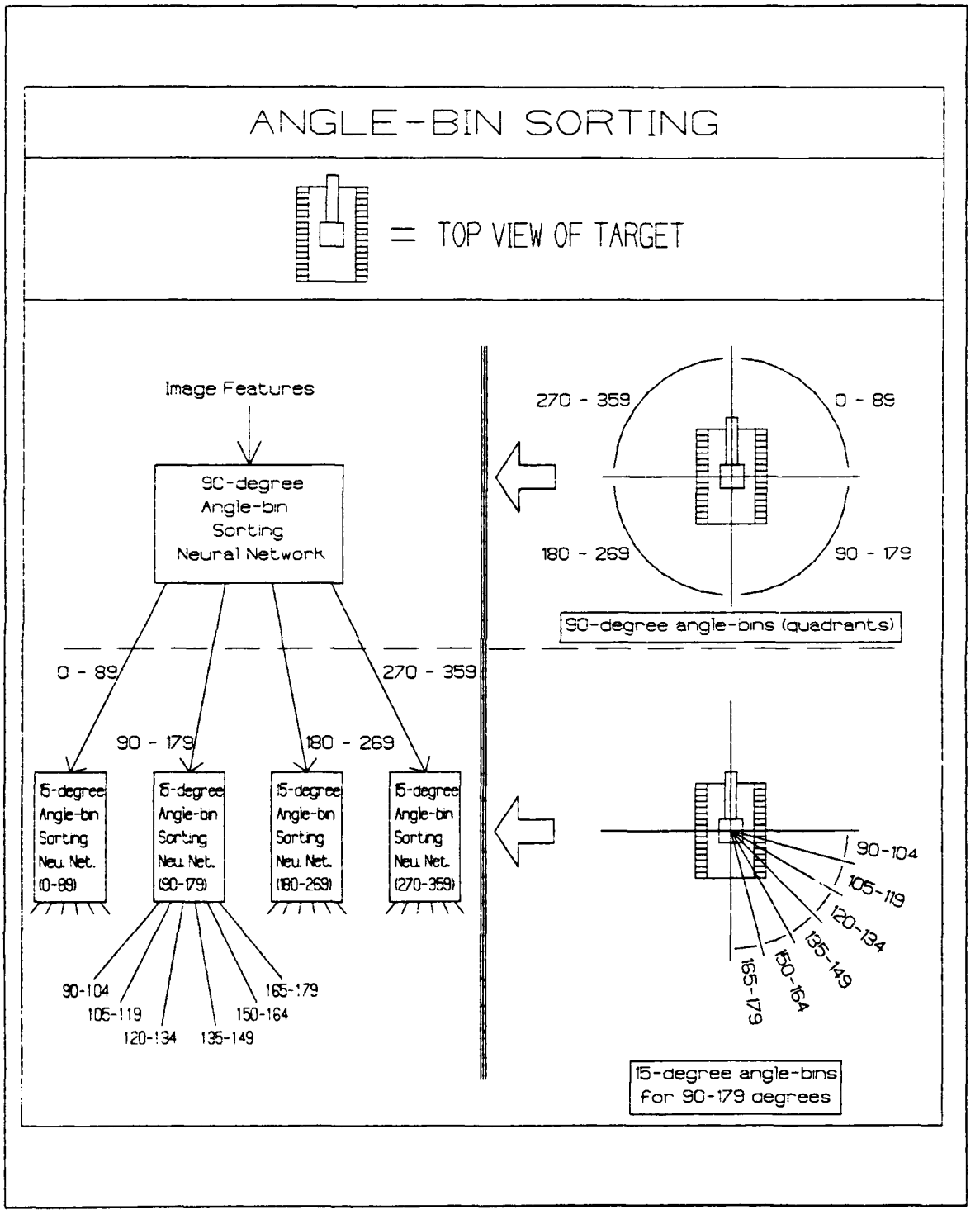


Figure 8. Determining the sensor view-angle of an object by sorting the object into consecutively smaller angle-bins.

angle bins; sorting the data into consecutively smaller angle-bins; and, finally, using one parameter (sensor viewing-angle) to determine pose of an object. Additionally, research details were provided. These include details on image generation, image manipulations, feature calculations, neural networks software/configurations, and the method for using the neural-network software to sort the data into the angle-bins.

IV. Results and Discussion

4.1 Introduction

This chapter provides and discusses the results obtained from this research. The chapter is divided into four major sections: a section discussing image generation with the modeling program, a section discussing the features selected for processing the imagery, a section discussing the application of these features to the synthetic IR imagery, and a section discussing the application of these features to real IR imagery. As will be shown, five features were selected which could be used to sort the synthetic imagery into the correct 15-degree angle-bin with an accuracy greater than 90%; however, when applied to real IR imagery, no conclusive results could be obtained.

4.2 Image Generation using SCNGEN.

4.2.1 Generating a White Background. For this research, the synthetic imagery was created with a white background. To create a white background (an option in the SCNGEN Scene Generation software), SCNGEN appears to force the background temperature to be hotter than the hottest portion of the model being generated. This ensures that the background will appear white and the target will appear dark.

4.2.1.1 Problems with Background and Silhouette Image. As can be seen in Figure 9, there are places in the silhouette imagery where the background appears to "leak" through. As a result, the silhouette image has small dots of white in the otherwise solid black silhouette. Though these minor dots do not radically alter the shape moments calculated from the silhouette, the dots may have a significant impact on intensity calculations—since they are hotter than they should be for their locations (see the next subsection for temperatures).

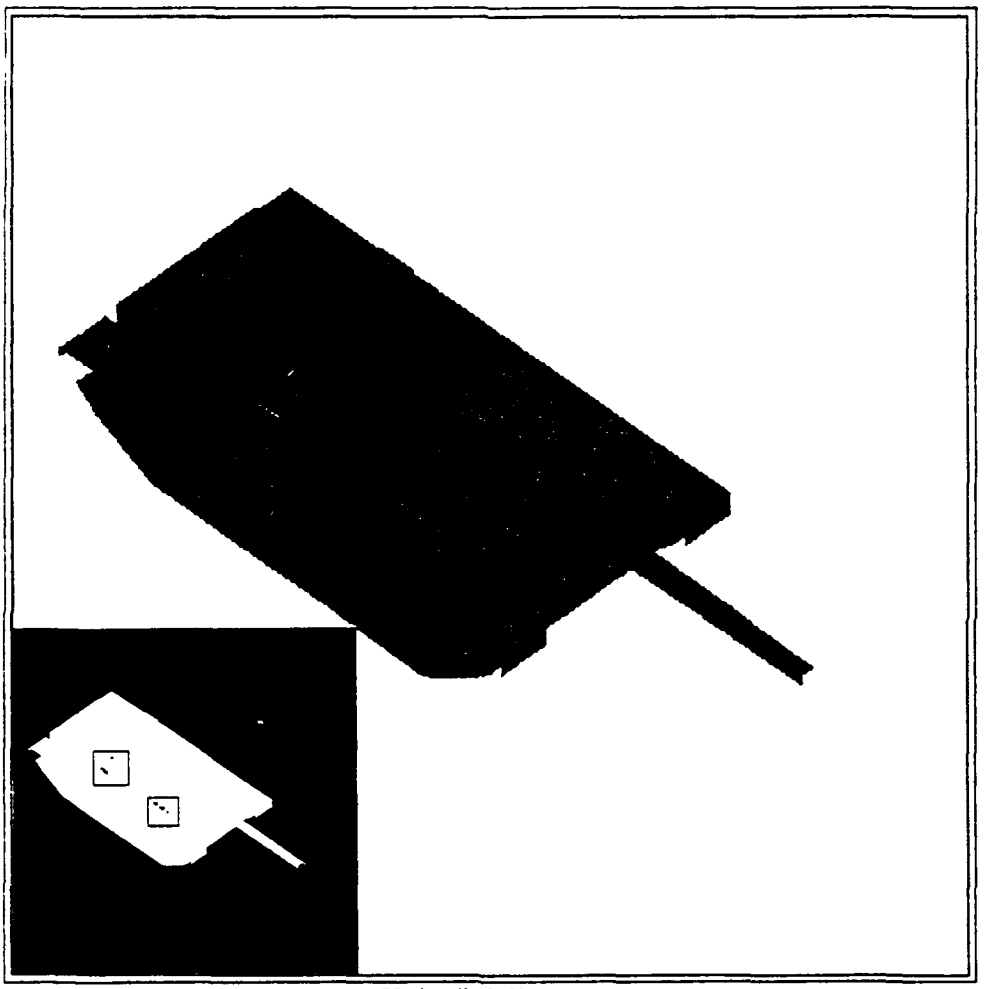


Figure 9. Example of background "leaking" through the silhouette image (the small white dots in the middle of the object—these are not artifacts from the photocopying process). The image shows where anomalous hot spots occur within the body of the object. The inset image is a reversed image with the anomalous spots more clearly identified.

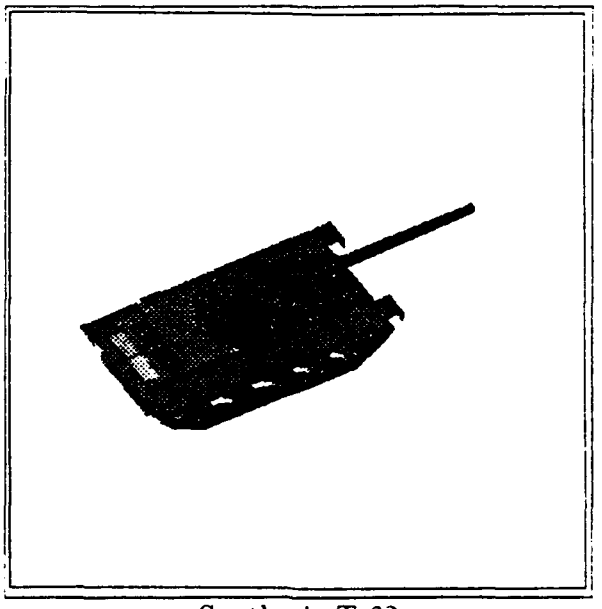
4.2.2 Visual Comparison to Actual IR-Imagery. As can be seen in comparing the synthetic IR imagery of a T-62 tank (Figure 10) with early-generation IR imagery and later-generation IR imagery, the synthetic IR imagery does not appear to truly reflect the "look" of a true IR image. This is due principally to the method by which the original floating-point image file (.IMG file) is converted to the gray-scale image (.DIS file).

The conversion was a straight linear conversion. The maximum floating-point value (M below) was assigned the hottest byte-value of "255" and the minimum floating-point (m below) value was assigned the coldest byte-value of "0." All floating-point values between these two extremes (F below) were assigned a byte-value (B below) by

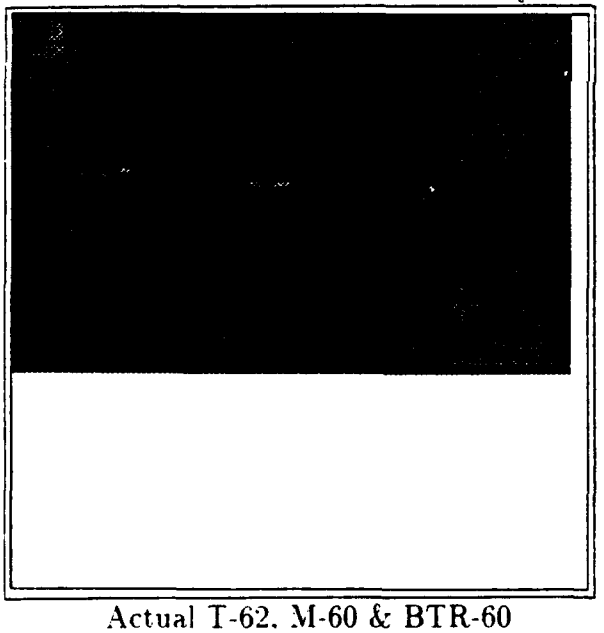
$$B = \frac{F - m}{M - m} * 255.$$

However, due to the way SCNGEN creates a white background (see the section above), the background was assigned a floating-point temperature of approximately 390 degrees Kelvin (hotter than any portion of the tank). The minimum temperature on the tank was approximately 280 degrees Kelvin. Since the temperatures of the vast majority of the surfaces on the tank were under 335 degrees Kelvin, the floating-point values of the tank were converted to relatively low byte-values (byte-values of 0 to 100—"cold" byte-values). This forces all but the direct engine compartments of the tank to appear cold (dark in the IR). It is possible to adjust the temperature range of the tank using RDIRMA on the .IMG file; however, since this research focused on the shape of the tank, no adjustments were required (i.e., no internal features such as shading or local maxima/minima were extracted).

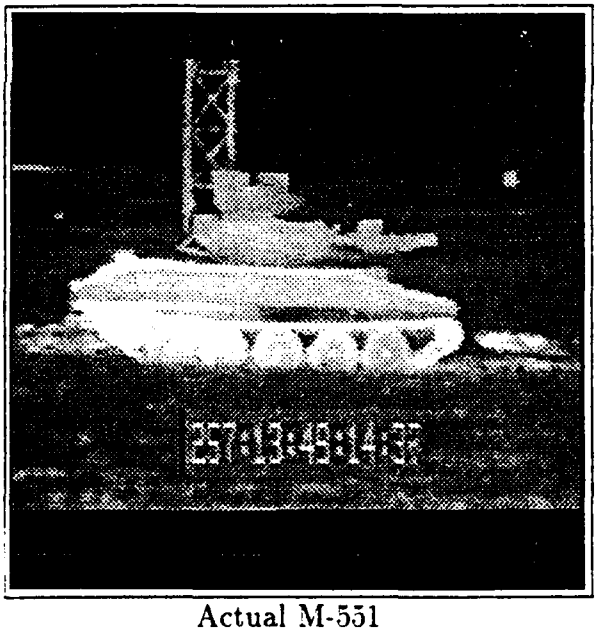
4.2.3 Small Depression-Angle Synthetic Imagery. As can be seen in Figure 13, for the model used in this research, SCNGEN synthetic imagery with a sensor depression-angle of zero degrees does not show the bottom of the tank tread. As a result, the silhouette and outline moments would not closely compare to actual IR imagery unless a significant amount of the tread were obscured in the actual IR imagery (and this wouldn't be a deterministic quantity).



Synthetic T-62



Actual T-62, M-60 & BTR-60



Actual M-551

Figure 10. TOP: Synthetic IR-Image of a Soviet T-62 main battle tank. BOTTOM LEFT: Early-generation IR image of actual Soviet T-62, USA M-60 and BTR-60. BOTTOM RIGHT: Later-generation IR image of actual USA M-551 Sheridan light tank

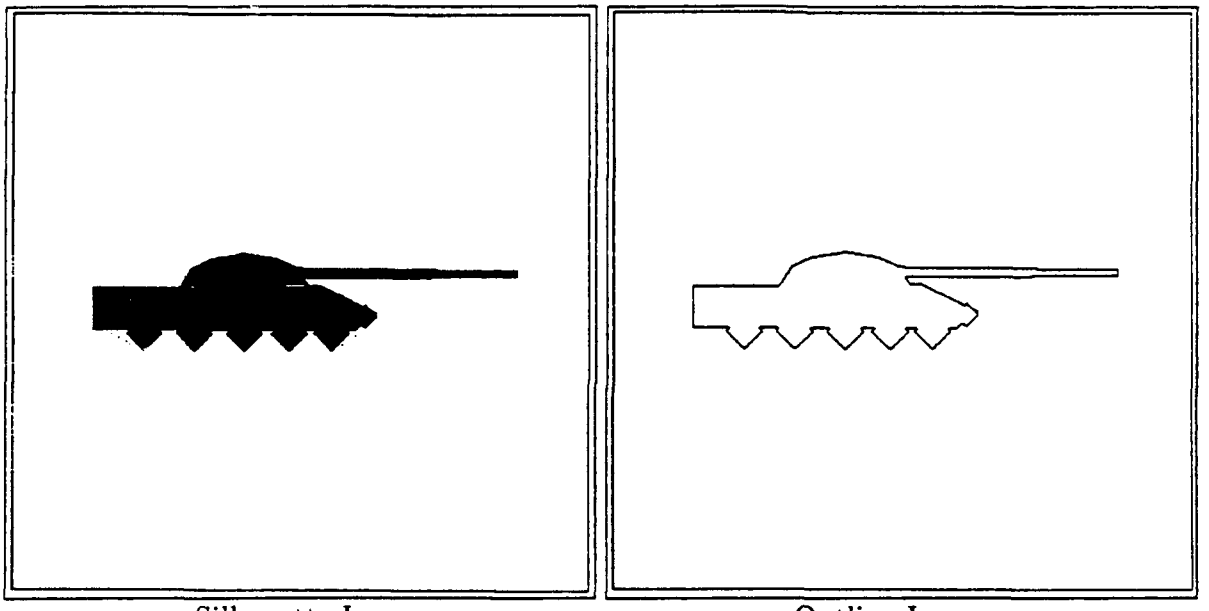


Figure 11. Corresponding silhouette and outline images of a T-62 with sensor depression-angle of 0 degrees and sensor view-angle of 90 degrees. Note the lack of tread across the bottom of the imagery.

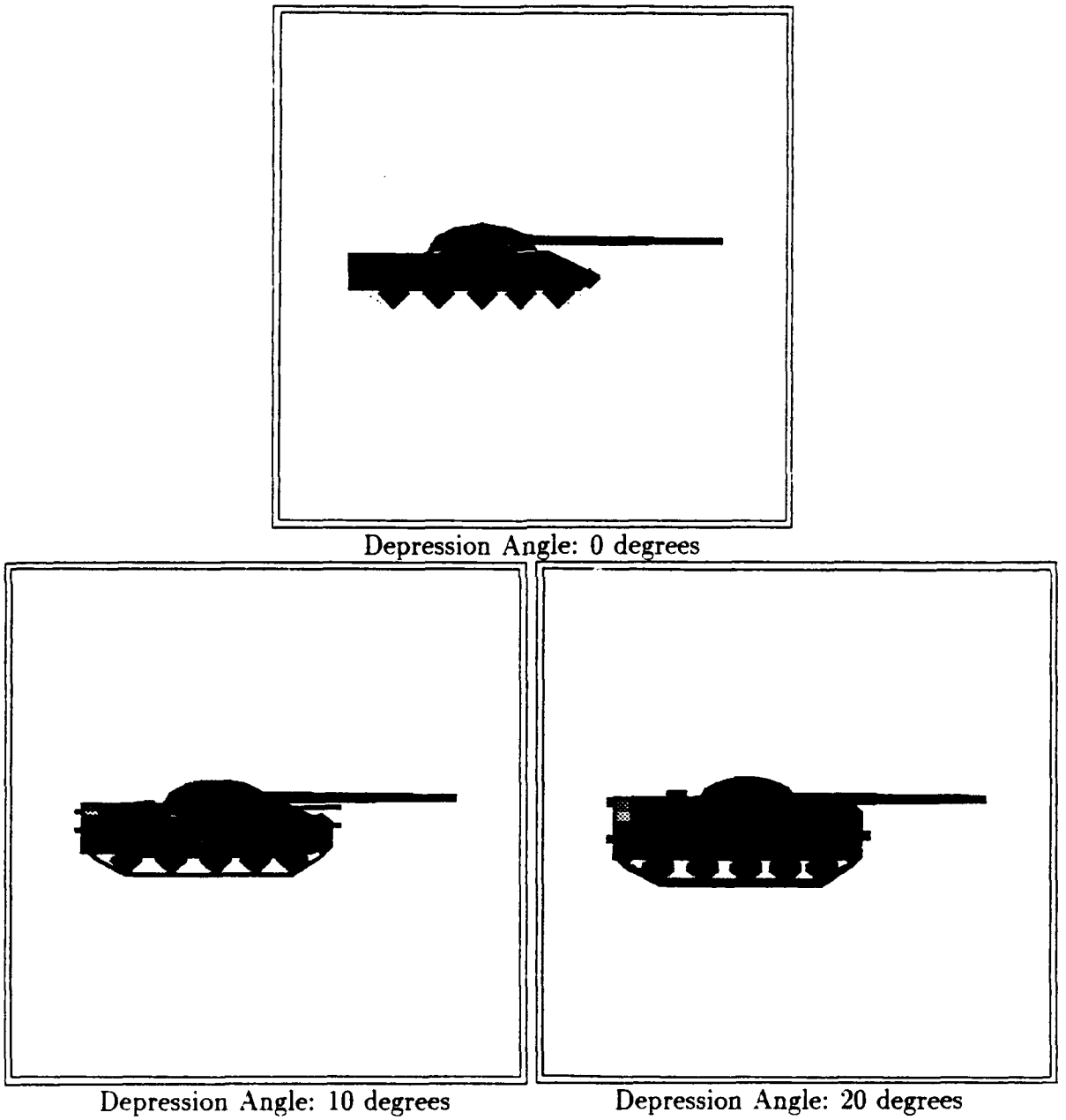
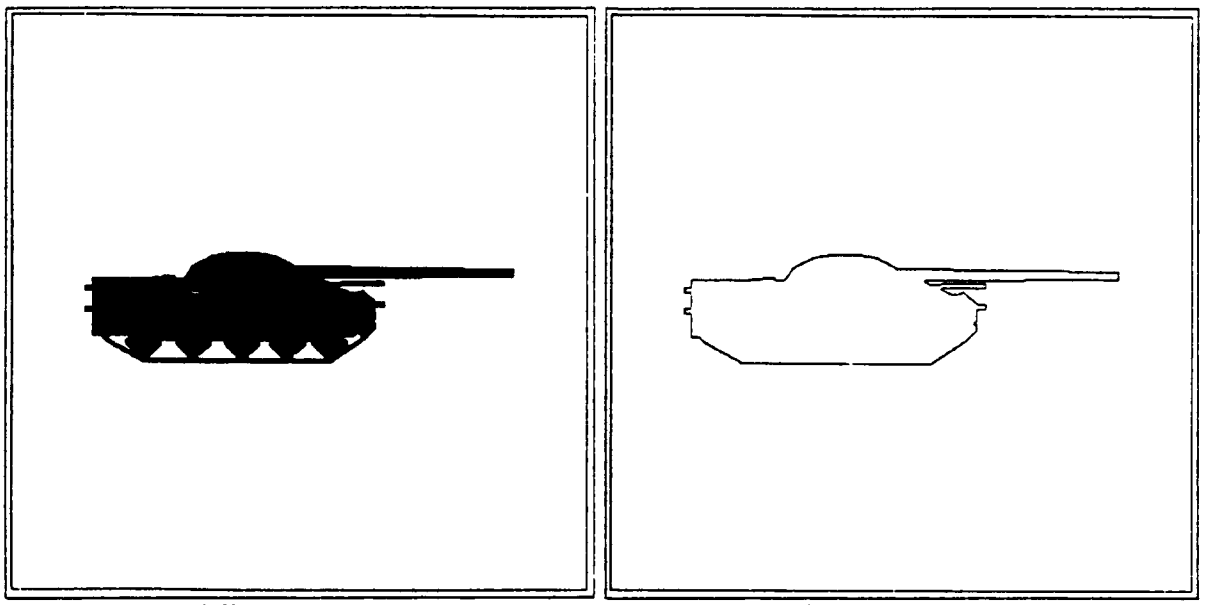


Figure 12. Synthetic T-62 images (sensor view-angle = 90 degrees for all three images).



Silhouette Image Outline Image

Figure 13. Corresponding silhouette and outline images of a T-62 with sensor depression-angle of 10 degrees and sensor view-angle of 90 degrees.

Furthermore, as can be also be seen in Figure 12, there is a significant amount of background visible through the tank tread/rollers area for side-view SCNGEN synthetic imagery with sensor depression-angles around 10 degrees. As a result of the methods used to calculate the silhouette and outline images, the outline image of the tank enclosed more area than the corresponding silhouette image of the tank at these small non-zero depression-angles (see Figure 13).

The binary silhouette image was generated by zeroing out all background numerical values and setting all non-background numerical values to one. Therefore, the silhouette image could contain "holes" where the background was visible. The binary outline image was generated by tracing the outline of the silhouette image. Therefore, provided the silhouette image did not have gaps in the tread areas between roller wheels, there should be no dramatic variations in the outline imagery as the sensor depression-angle is varied past the zero-degree sensor depression-angle.

Consequently, though, to use the silhouette moments for accurate comparison between the SCNGEN synthetic IR imagery and actual IR imagery, depression angles greater than 20 or 30 degrees would be more useful than smaller depression-angles. To use the outline moments for accurate comparison between synthetic and actual IR imagery, it would be best to avoid the very small sensor-depression-angles around zero degrees.

4.3 Features.

Thirty-six features were initially evaluated to determine which features could yield the most useful sensor orientation information. These features were calculated using the silhouette and the outline moments.

For the silhouette moments, the features were numbered:

$$feature_0 = M_{20}/M_{02}$$

$$feature_1 = M_{20}/M_{11}$$

$$feature_2 = M_{02}/M_{11}$$

$$feature_3 = M_{30}/M_{03}$$

$$feature_4 = M_{30}/M_{21}$$

$$feature_5 = M_{03}/M_{21}$$

$$feature_6 = M_{30}/M_{12}$$

$$feature_7 = M_{03}/M_{12}$$

$$feature_8 = M_{12}/M_{21}$$

$$feature_9 = M_{20}/M_{00}$$

$$feature_{10} = M_{02}/M_{00}$$

$$feature_{11} = M_{11}/M_{00}$$

$$feature_{12} = M_{30}/(M_{00})^2$$

$$feature_{13} = M_{03}/(M_{00})^2$$

$$feature_{14} = M_{21}/(M_{00})^2$$

$$feature_{15} = M_{12}/(M_{00})^2$$

where the M represents the moments calculated from the silhouette images. Feature 16 was the orientation angle of the silhouette image as calculated in Chapter 3. Feature 15 was the overall maximum height-to-width ratio.

For the outline moments, the features were numbered:

$$feature_{18} = M_{20}/M_{02}$$

$$feature_{19} = M_{20}/M_{11}$$

$$feature_{20} = M_{02}/M_{11}$$

$$feature_{21} = M_{30}/M_{03}$$

$$feature_{22} = M_{30}/M_{21}$$

$$feature_{23} = M_{03}/M_{21}$$

$$feature_{24} = M_{30}/M_{12}$$

$$feature_{25} = M_{03}/M_{12}$$

$$feature_{26} = M_{12}/M_{21}$$

$$feature_{27} = M_{20}/M_{00}$$

$$feature_{28} = M_{02}/M_{00}$$

$$feature_{29} = M_{11}/M_{00}$$

$$feature_{30} = M_{30}/(M_{00})^2$$

$$feature_{31} = M_{03}/(M_{00})^2$$

$$feature_{32} = M_{21}/(M_{00})^2$$

$$feature_{33} = M_{12}/(M_{00})^2$$

where the M represents the moments calculated from the outline images. Feature 34 was the orientation angle of the outline image as calculated in Chapter 3. Feature 35 was the height-to-width ratio calculated using the images' maximum vertical and horizontal slices (see Chapter 3).

4.3.1 Dependence on Sensor Angles. To determine which features would yield the most information about the orientation of the target within the synthetic imagery, 216 images of the T-62 were generated. These images were generated for sensor depression-angles of 10, 20, 30, 40, 50, and 60 degrees with sensor view-angles every 10 degrees starting with 0 degrees and ending with 350 degrees (see Figure 14).

The data were then plotted (see Figures 17 through to 21 for plots of the five selected features; see the Appendices for the plots of all 36 features). After evaluating these plots, five features were selected which appeared to fully represent all 36 features. The rationale for selecting the five features is provided in Section 4.3.2.2.

4.3.2 The Five Features Selected. The five features selected for this research were $feature_0$, $feature_1$, $feature_{19}$, $feature_{22}$, and $feature_{31}$. These features were selected with the intent to discriminate among the angle-bins for the sensor view-angles. As such, the five features selected appeared to have different data profiles (e.g., rate of change, relative magnitudes, etc.) depending on the side or quadrant from which the target was viewed (Figures 17 - 21). The intent was to allow the targets to be sorted into quadrants using

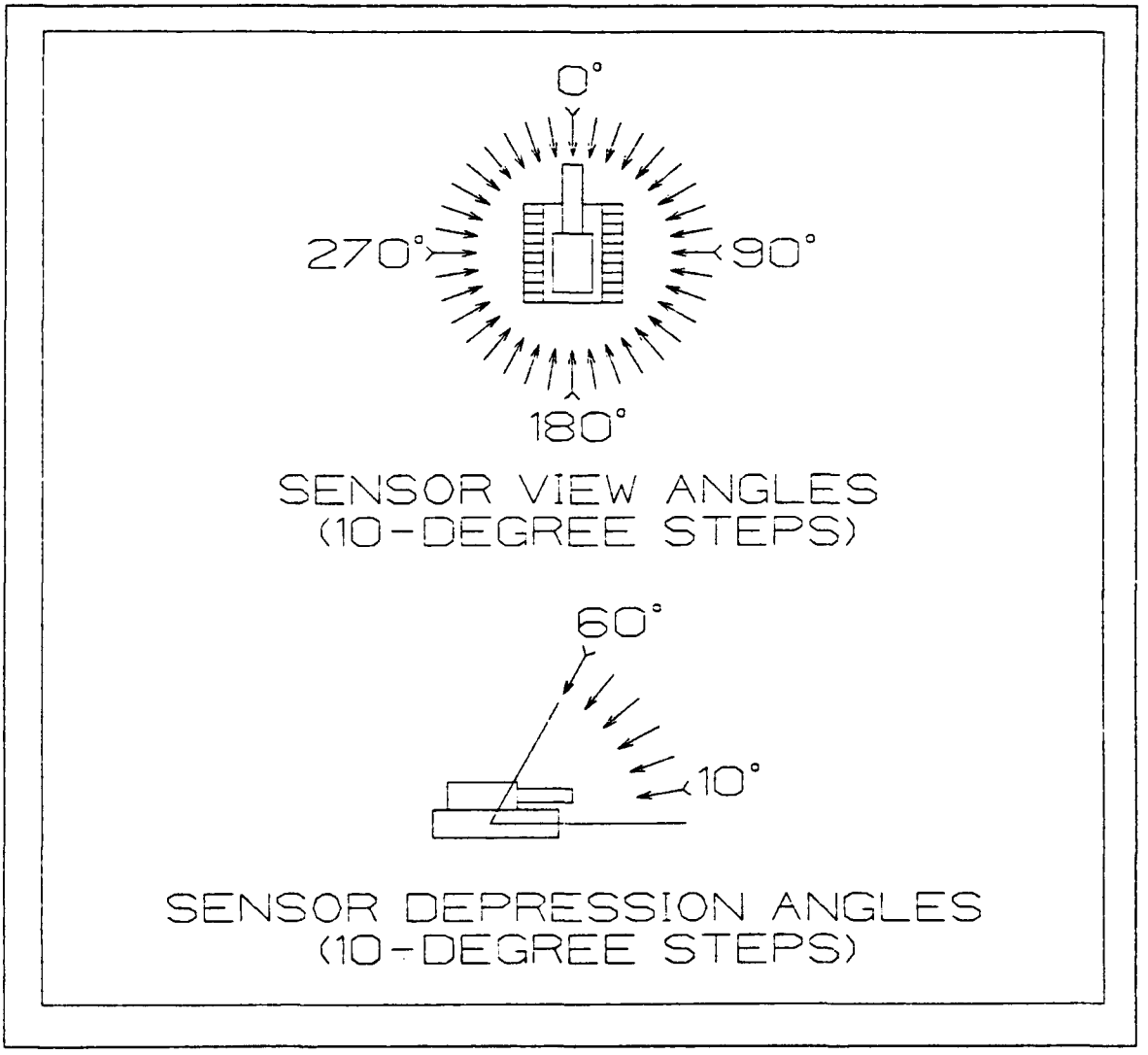


Figure 14. Views for which the features were evaluated (each arrow represents a view point). For each of the six sensor depression-angles, the 36 sensor view-angles were used to create imagery (resulting in 216 images).

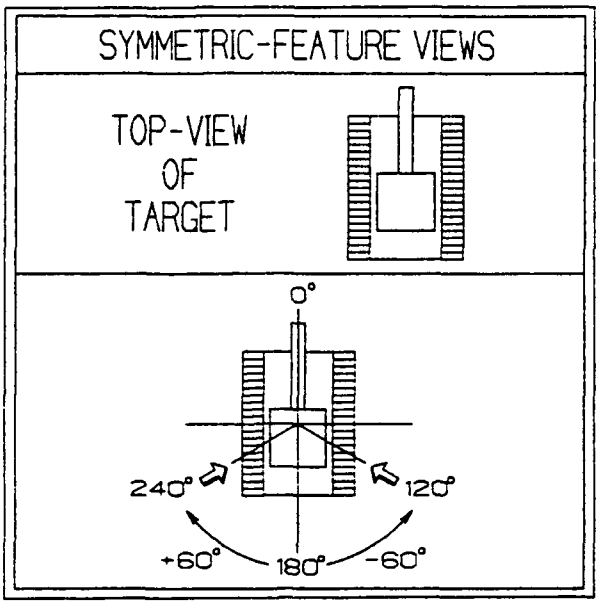
the five features. Further sorting would be accomplished using the same features or using a separate set of features (if required).

4.3.2.1 Symmetry and Asymmetry Properties. In general, the features were either fully symmetric or significantly asymmetric about the 180-degree sensor view-angle point (see plots of features: Figure 17 through to Figure 21). For the symmetric features, two objects rotated an equivalent amount from the 180-degree sensor view-angle point would yield similar numerical values (see the tanks in Figure 15—one figure is rotate 60 degrees clockwise from the 180-degree sensor view-angle point; the other is rotated 60 degrees counterclockwise from the 180-degree sensor view-angle point). For the asymmetric features, two objects rotated 180 degrees from one another would generally yield relatively close numerical values (see the tanks in Figure 16—one tank is at a sensor view-angle of 120 degrees; the other is at a sensor view-angle of 120 + 180 degrees).

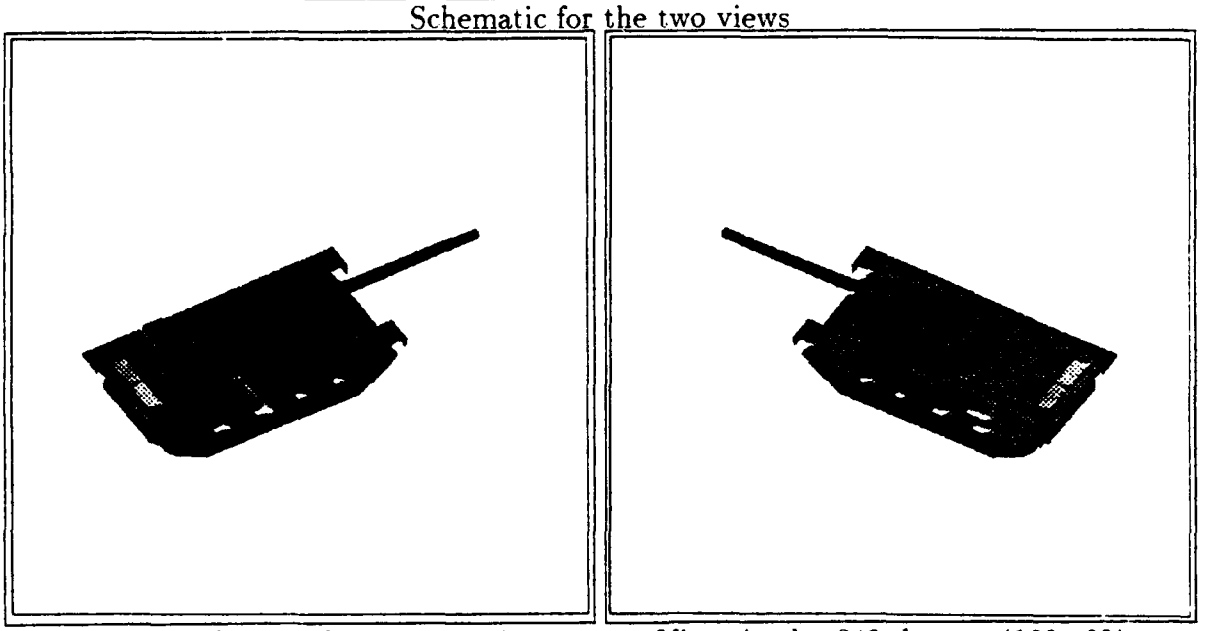
Both the symmetric and asymmetric features tended to have local maxima or minima occurring near or at the quadrant transition points (i.e., sensor view-angles of 0, 90, 180, and 270 degrees). Also, the asymmetric features had a sinusoidal appearance in the data with respect to changes in the sensor view-angle.

NOTE: The top left corner of the image was the origin ($x = 0, y = 0$). Therefore, portions of the tank to the right of the tank's center yielded larger numerical values than portions to the left. As such, placement of the barrel and turret could be credited with the differences in the magnitudes of the maxima and minima for the asymmetric features since the barrel/turret pointing to the right would yield higher values than the barrel/turret pointing to the left.

4.3.2.2 Rationale for Selecting the Five Features. Rationale for selecting each of the five features is given below:



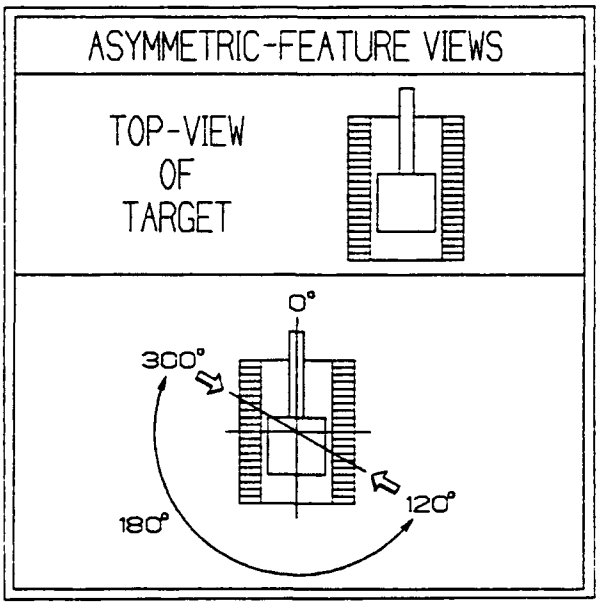
Schematic for the two views



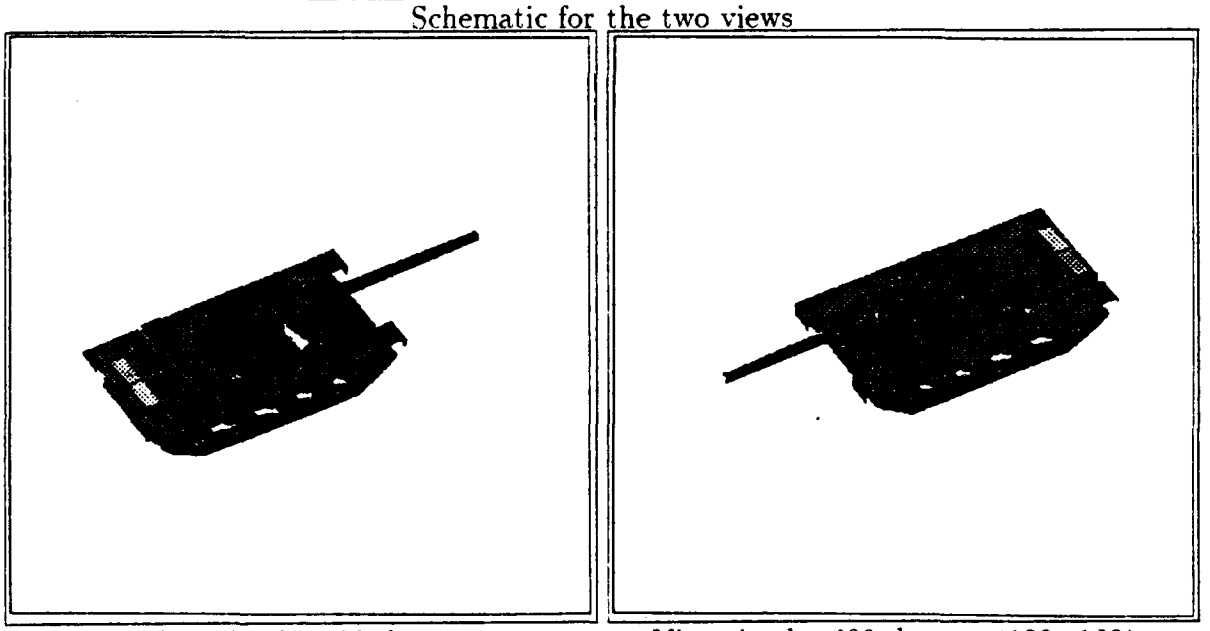
View Angle: 120 degrees (180-60)

View Angle: 240 degrees (180+60)

Figure 15. Synthetic T-62 imagery which yield similar values for the symmetrical features. These two images are at sensor view-angles of 180±60 degrees.



Schematic for the two views



View Angle: 120 degrees

View Angle: 300 degrees (120+180)

Figure 16. Synthetic T-62 imagery which yield similar values for the asymmetrical features. These two images are separated by a sensor view-angles of 180 degrees.

- Feature 0 (silhouette moments: M_{20}/M_{02}): Symmetric about the 180-degree sensor view-angle point with peaks at the quadrant transition points. It is representative of the majority of the symmetric features (see Figure 17).
- Feature 1 (silhouette moments: M_{20}/M_{11}): Asymmetric about the 180-degree sensor view-angle point with peaks offset 20 degrees or so from the quadrant transition points. It is representative of the majority of the asymmetric features (see Figure 18).
- Feature 19 (outline moments: M_{20}/M_{11}): Asymmetric about the 180-degree sensor view-angle point with peaks offset 20 degrees or so from the quadrant transition points. The data rate-of-change varied differently about the 90-degree and 270-degree quadrant transition points (see Figure 19).
- Feature 22 (outline moments: M_{30}/M_{21}): Asymmetric about the 180-degree sensor view-angle point with peaks offset 50 degrees or so from the quadrant transition points. The relative data magnitudes of the peaks differed (see Figure 20).
- Feature 31 (outline moments: $M_{03}/(M_{00})^2$): Symmetric about the 180-degree sensor view-angle point. However, the peaks did not all occur at the quadrant transition points. Notably, there were two peaks (minima, in this case) at the 120-degree and 240-degree sensor view-angle points (but only for the higher sensor depression-angles such as 30 degrees and higher—the distinction of this feature disappeared at the lower sensor depression-angles such as 10 degrees) (see Figure 21).

4.3.3 Testing the Selected Features' Invariance. To ensure the features were shift and scale invariant, the numerical values were extracted and compared for images which differed only in the placement or size of the target in the field of view.

4.3.3.1 Shift Invariance. The shift invariance of the features was confirmed by extracting features and comparing the results from imagery such as shown in Figure 22. There were no differences in the features; therefore, the algorithm used was shift invariant.

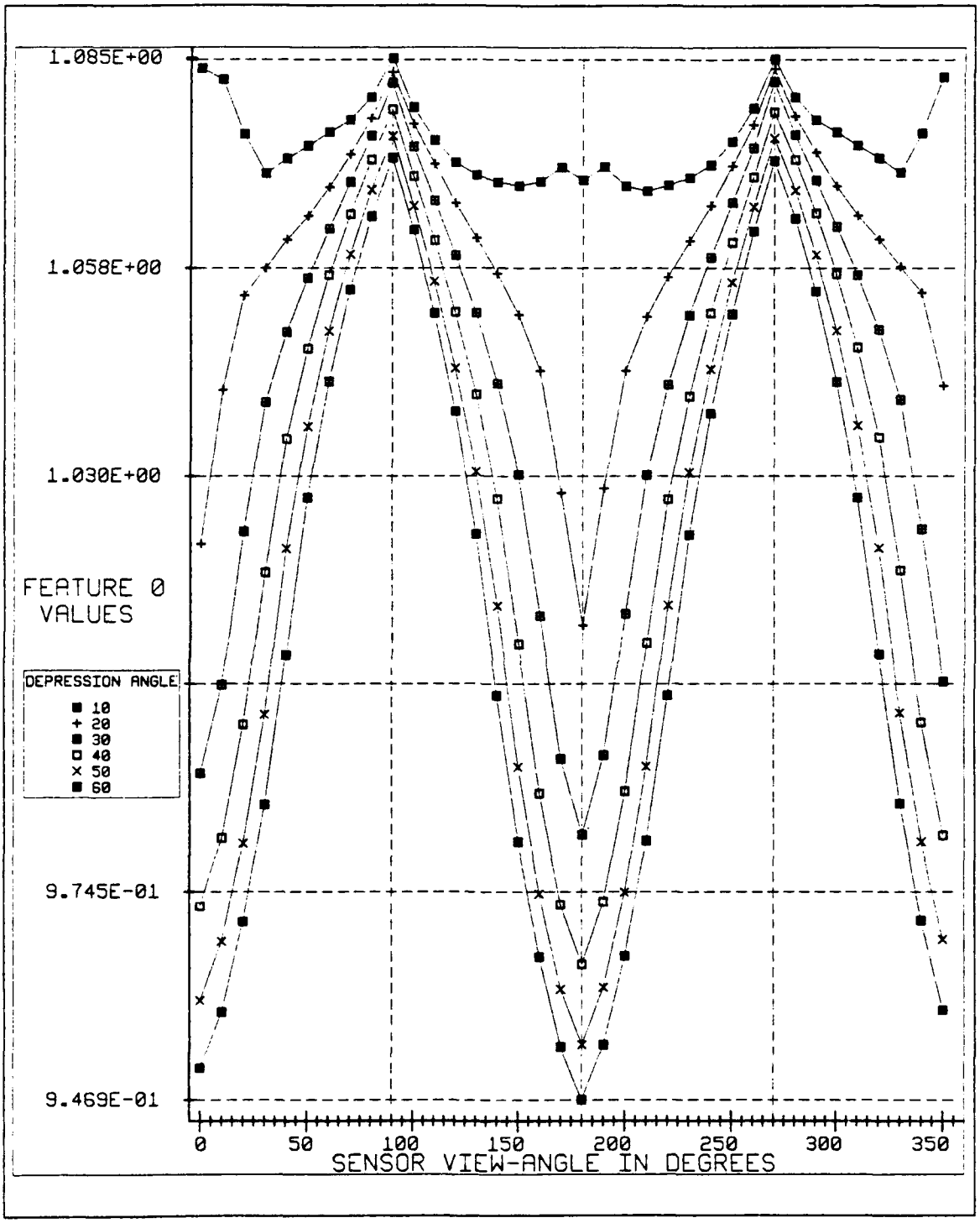


Figure 17. Feature 0 (silhouette moments: M_{20}/M_{02}) plotted for the six sensor depression-angles and the 36 sensor view-angles.

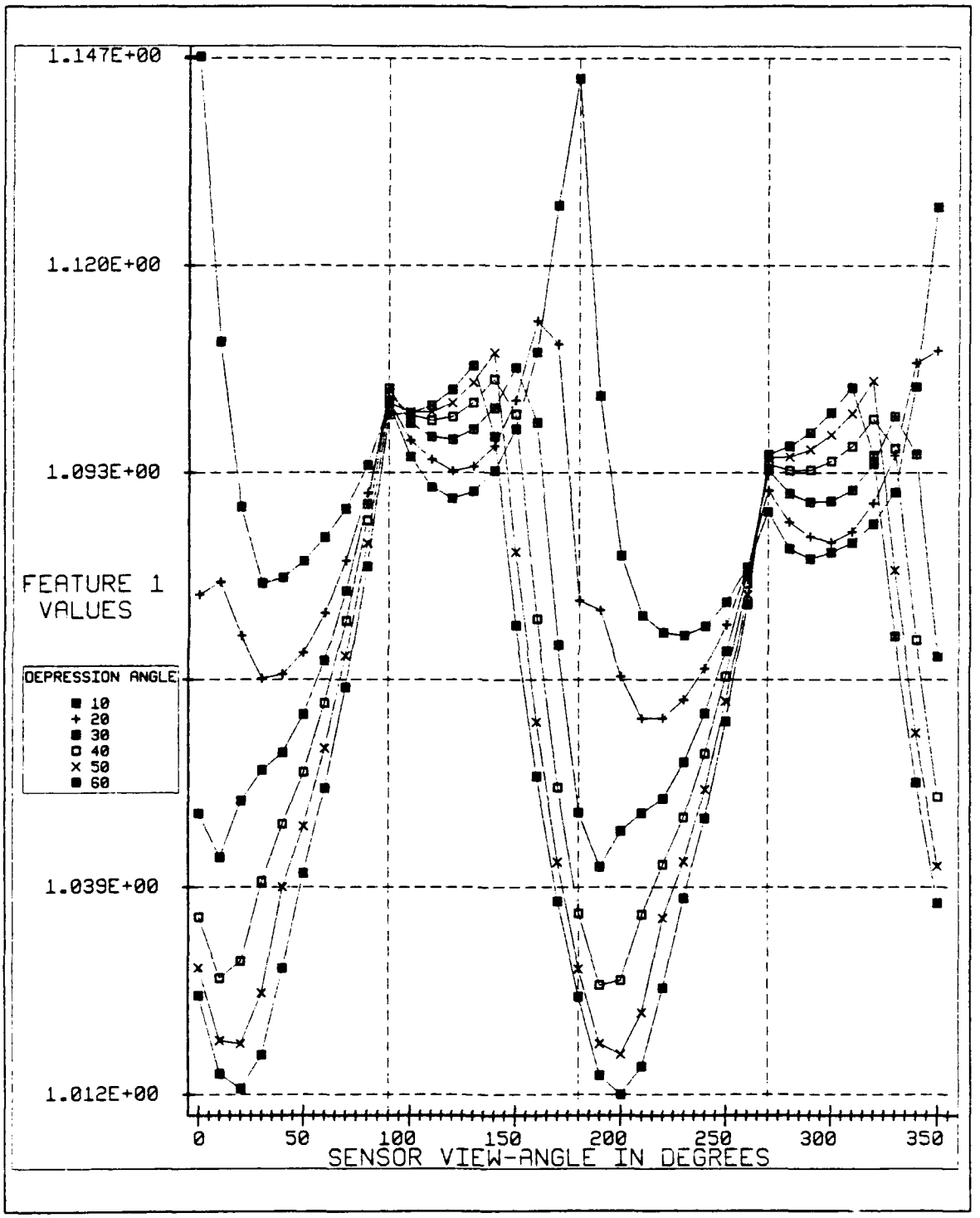


Figure 18. Feature 1 (silhouette moments: M_{20}/M_{11}) plotted for the six sensor depression-angles and the 36 sensor view-angles.

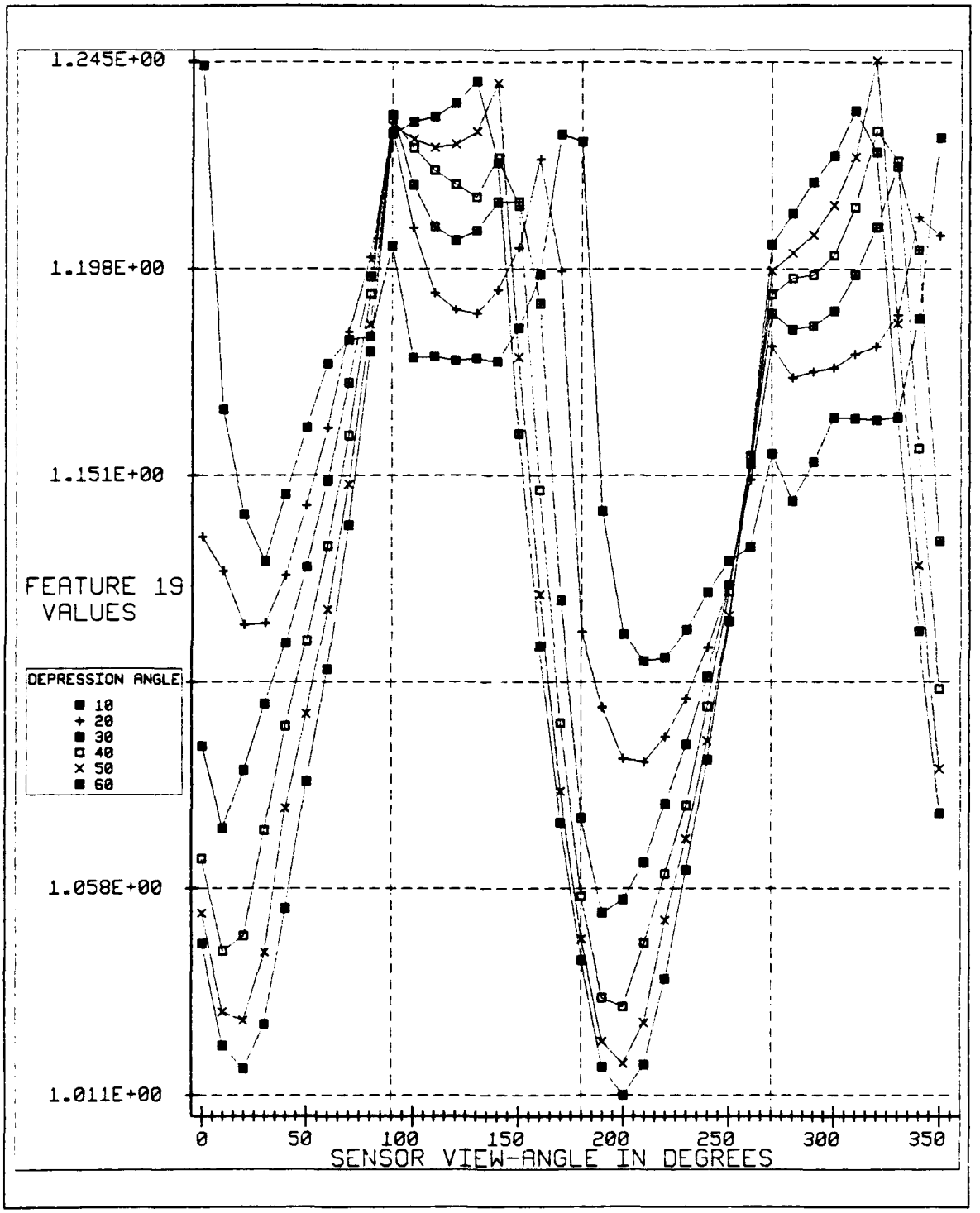


Figure 19. Feature 19 (outline moments: M_{20}/M_{11}) plotted for the six sensor depression-angles and the 36 sensor view-angles.

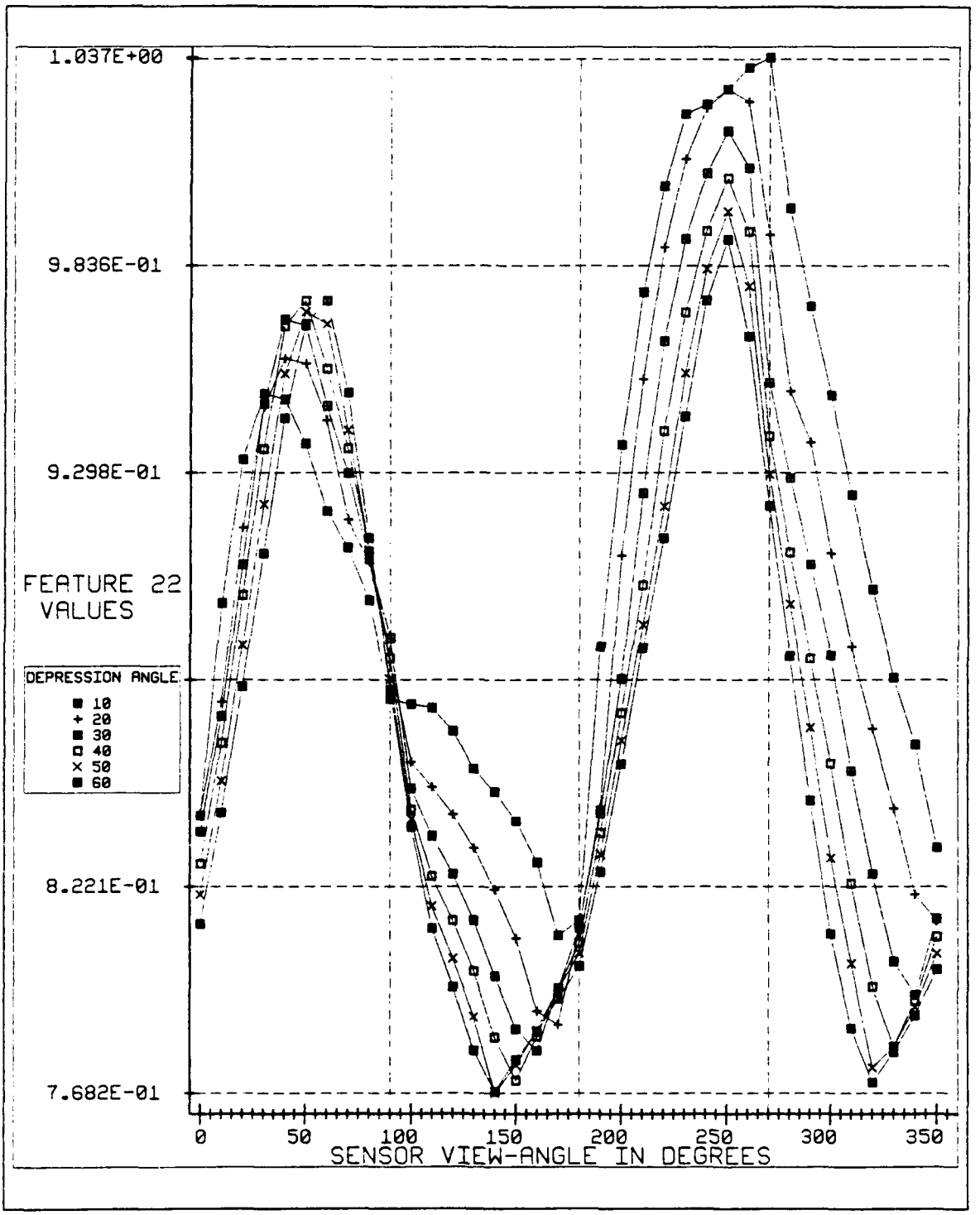


Figure 20. Feature 22 (outline moments: M_{30}/M_{21}) plotted for the six sensor depression-angles and the 36 sensor view-angles.

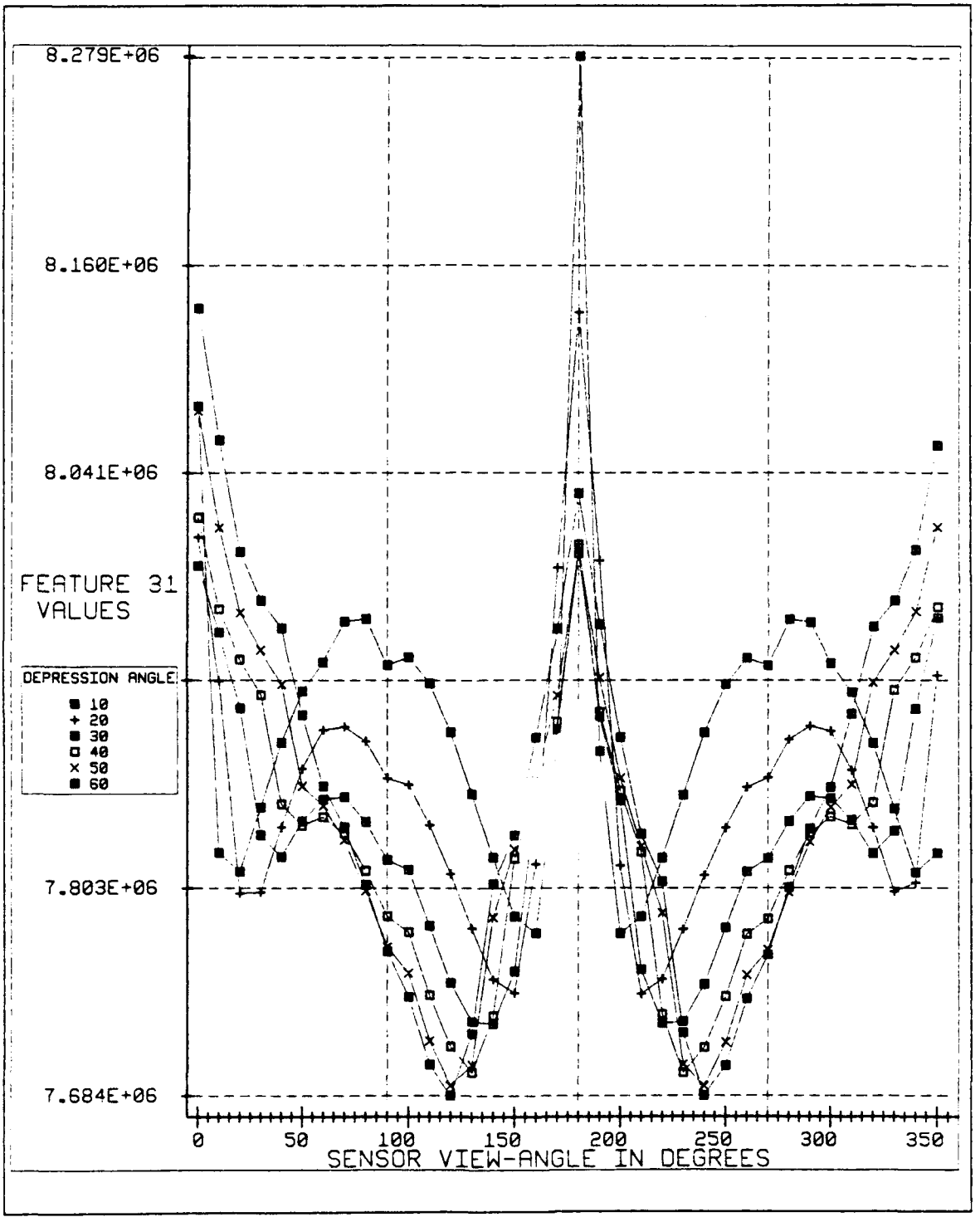
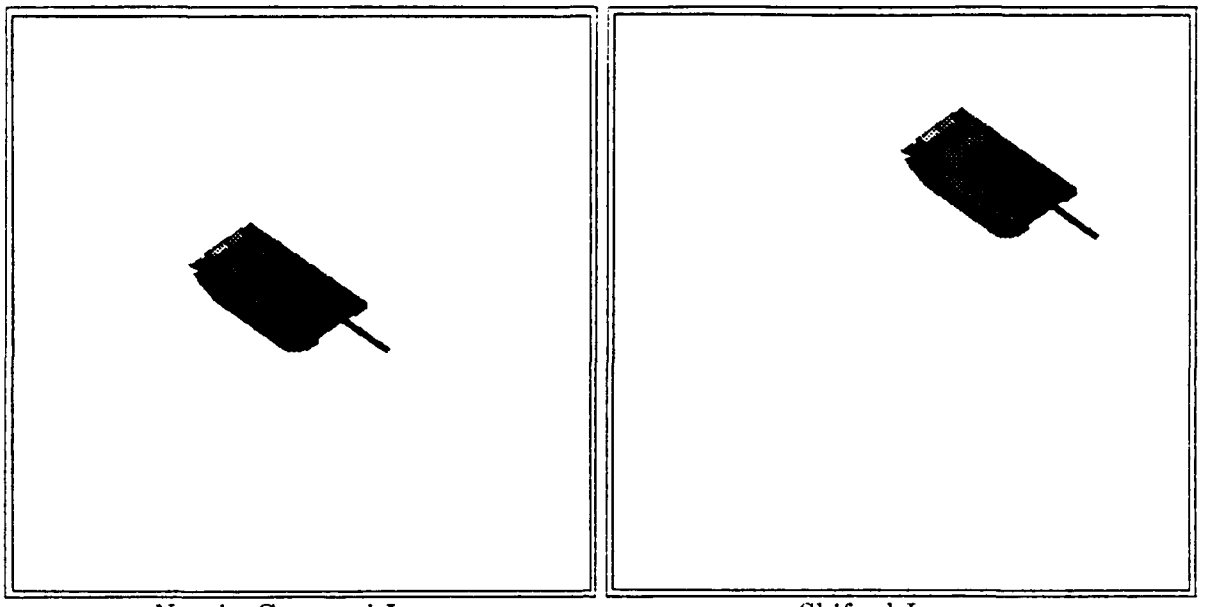


Figure 21. Feature 31 (outline moments: $M_{03}/(M_{00})^2$) plotted for the six sensor depression-angles and the 36 sensor view-angles.

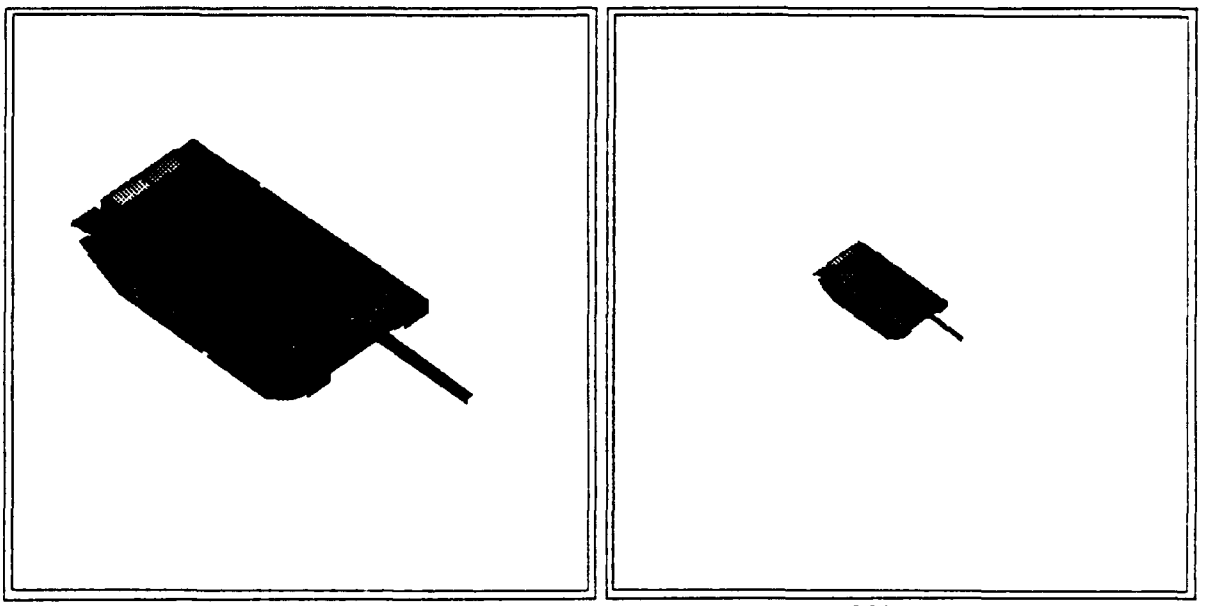


Nearly-Centered Image Shifted Image

Figure 22. Imagery used for testing shift invariance of features.

4.3.3.2 Scale Invariance. The sensitivity of the features to changes in scale was explored for imagery ranging from a FOV setting of 0.80 (the image fills 80% of the FOV) to a FOV setting of 0.30 (see Figure 23). Results for the five selected features are below (see Table 1); results for all 36 features are in the appendices. Figure 24 shows the five selected features plotted with respect to the changes in the FOV.

As can be seen, the features were not perfectly scale invariant; however, the change in scale between the 0.80 FOV and 0.30 FOV imagery is dramatic and represents a major change in effective range to the target. For comparison purposes, though, the features were relatively tolerant of minor changes in scale (compare 0.80 FOV data to 0.75 FOV data). As a result, the approximate range to the target would be required in actual IR imagery so as to predict the approximate FOV setting to use in the synthetic imagery. This range data would be available through the use of LASER RADAR, or, perhaps, conventional RADAR, etc. Also, these five features are well behaved and could be represented or approximated by



FOV: 0.80 FOV: 0.30

Figure 23. Range of image size use for testing scale invariance of features.

Table 1. Test of Scale Invariance. Data is also plotted in the following figure.

FOV		Feature 0	Feature 1	Feature 19	Feature 22	Feature 31
0.80		1.039459	1.052680	1.106960	0.968403	7876061.5
0.75		1.034895	1.046481	1.095112	0.971127	7857673.0
0.70		1.030558	1.040628	1.083212	0.973176	7848011.0
0.65		1.026416	1.035109	1.072552	0.975585	7841854.0
0.60		1.022625	1.030011	1.061605	0.976429	7845906.0
0.55		1.019457	1.025668	1.052679	0.978594	7857062.5
0.50		1.016218	1.021325	1.043655	0.980153	7874587.0
0.45		1.013145	1.017298	1.035349	0.981804	7897358.5
0.40		1.010445	1.013731	1.028165	0.984246	7929890.0
0.35		1.008149	1.010651	1.021743	0.986777	7968558.0
0.30		1.006068	1.007895	1.016129	0.988331	8017122.5

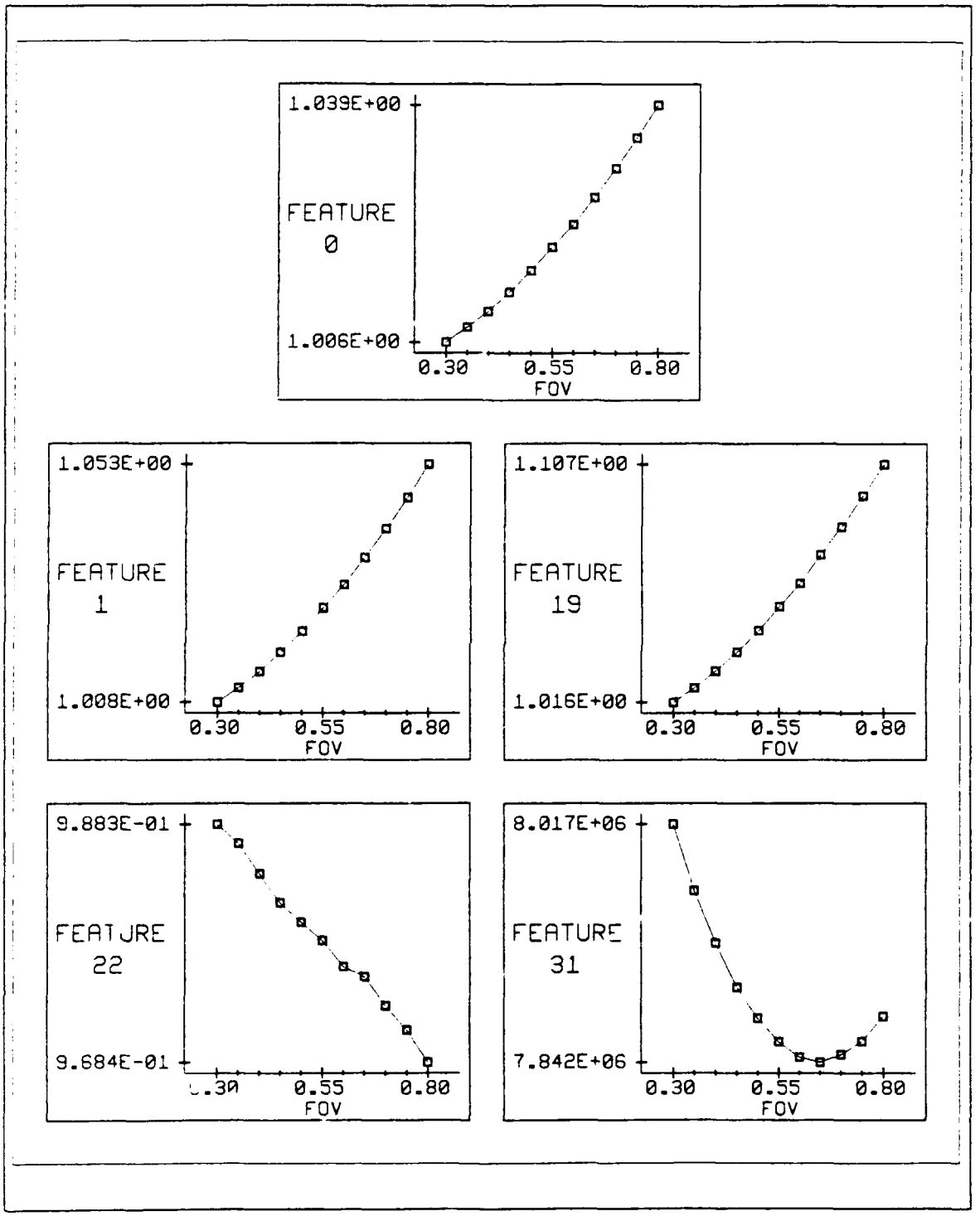


Figure 24. Plot of the five selected features with respect to changes in the field of view (FOV). Note that all are fairly well behaved and could be represented or approximated with linear, quadratic, or cubic functions.

linear, quadratic or cubic equations (thus allowing for some degree of normalization of the feature value given the FOV of the object).

4.4 Determining Sensor View-Angle for Synthetic IR-Imagery.

This section discusses how well the sensor view-angle was determined for the synthetic T-62 imagery. As will be seen, given both the sensor depression-angle and the target roll-angle, the sensor view-angle could be determined within 7.5 degrees of its actual value (this is a result of placing the object into the correct 15-degree angle-bin as discussed in Chapter 3 and using the center image as the reference image). Better accuracy may have been possible by further sorting the imagery into the correct 5-degree angle-bin; however, this was not explored.

NOTE: For this research, a sensor depression-angle of 40 degrees was used for the synthetic imagery. This was chosen arbitrarily, but predicated on the constraints of Section 4.2.3.

4.4.1 Hierarchical Sorting into Angle-Bins. Without sorting the imagery into consecutively smaller angle-bins, the accuracy obtainable was ≈60%. With sorting into consecutively smaller angle-bins, the accuracy increased to slightly better than 90%. This number was not dramatically dependent upon the path taken (see Table 2). Within Table, "path" refers to the manner in which the imagery was sorted into consecutively smaller angle-bins. For example, $360 \rightsquigarrow 15$ means the imagery was sorted directly into one of the 24 15-degree angle-bins given the entire 360-degree spectrum (without any "hierarchical sorting"); $360 \rightsquigarrow 90 \rightsquigarrow 45 \rightsquigarrow 15$ means the imagery was first sorted into one of the four 90-degree angle-bins, then was sorted into one of the two 45-degree angle-bins within the 90-degree angle-bin, and then was sorted into one of the three 15-degree angle-bins within the 45-degree angle-bin.

4.4.2 Dependence on Sensor Depression-Angle. The accuracy shown in the previous section depends on accurate knowledge of the sensor depression-angle and the target roll-

Table 2. Accuracy of 15-degree angle-bin determination for 40-degree sensor depression-angle data. Where appropriate, average values are given.

Path (Consecutively smaller angle-bins)	Accuracy (Train/Test Percentages)
$360 \rightsquigarrow 15$	58.7 / 56.6
$360 \rightsquigarrow 90 \rightsquigarrow 45 \rightsquigarrow 15$	99.1 / 92.7
$360 \rightsquigarrow 90 \rightsquigarrow 15$	99.1 / 91.8
$360 \rightsquigarrow 45 \rightsquigarrow 15$	92.5 / 92.5

Table 3. Accuracy of 15-degree angle-bin determination for training with 40-degree sensor depression-angle data and testing with 30-degree sensor depression-angle data. Average values are given.

Path (Consecutively smaller angle-bins)	Accuracy (Train/Test Percentages)
$360 \rightsquigarrow 180 \rightsquigarrow 90 \rightsquigarrow 45 \rightsquigarrow 15$	96.1 / 44.6
$360 \rightsquigarrow 90 \rightsquigarrow 45 \rightsquigarrow 15$	96.5 / 44.4
$360 \rightsquigarrow 45 \rightsquigarrow 15$	90.0 / 43.4

angle (both can be determined using LASER RADAR, etc). To test the robustness of the features and the technique of angle-bin sorting, the neural network used in this research was configured so that it trained using the 40-degree sensor depression-angle data while testing with the 30- and 50-degree sensor depression-angle data. The results are given in Tables 3 and 4 (see Section 4.4.1 for details on the notation used in the tables). As can be seen, the accuracy still is not dramatically dependent upon the path taken; however, the results are significantly less when this "cross training & testing" is used than when a single sensor depression-angle is used.

4.4.3 Discussion. The five selected features show promise for determining the sensor view-angle given the sensor depression-angle and the target roll-angle. However, as shown in

Table 4. Accuracy of 15-degree angle-bin determination for training with 40-degree sensor depression-angle data and testing with 50-degree sensor depression-angle data. Average values are given.

Path (Consecutively smaller angle-bins)	Accuracy (Train/Test Percentages)
$360 \rightsquigarrow 180 \rightsquigarrow 90 \rightsquigarrow 45 \rightsquigarrow 15$	94.1 / 67.6
$360 \rightsquigarrow 90 \rightsquigarrow 45 \rightsquigarrow 15$	96.0 / 68.9
$360 \rightsquigarrow 45 \rightsquigarrow 15$	87.8 / 67.2

the previous section, the sensor depression-angle must be known well within 10 degrees of its actual value or the accuracy of the angle-bin sorting technique drops off dramatically. The features are more tolerant of errors in the sensor depression-angle which lead to a slightly larger value than actually present. This is illustrated in the previous section where the 50-degree data was accurately placed in the correct 15-degree angle-bin $\approx$70% of the time (as compared to $\approx$45% of the time for the 30-degree data).

4.5 Determining Sensor View-Angle for Actual IR-Imagery.

4.5.1 Early-Generation IR-Sensor Imagery. This section compares the values measured from the synthetic T-62 image to objects in actual IR imagery. In this section, the imagery was obtained using early-generation IR sensor technology. The imagery (see Figure 25) is of a T-62 tank, an M-60 tank, and a BTR-60. To facilitate human viewing of the imagery, a histogram-adjusted version of the same image is in Figure 26.

4.5.1.1 Problems. The imagery was taken at a range of 2010 meters. Therefore, the objects in the imagery appear small—too small to easily distinguish the T-62 from the M-60. The middle object appears to be larger and appears to have a cupola (a metal basket-shaped structure mounted over the tank turret hatch from which the tank commander can view the battlefield, etc.). These two factors suggest the middle object is the M-60. The object on the far left appears to be the T-62.

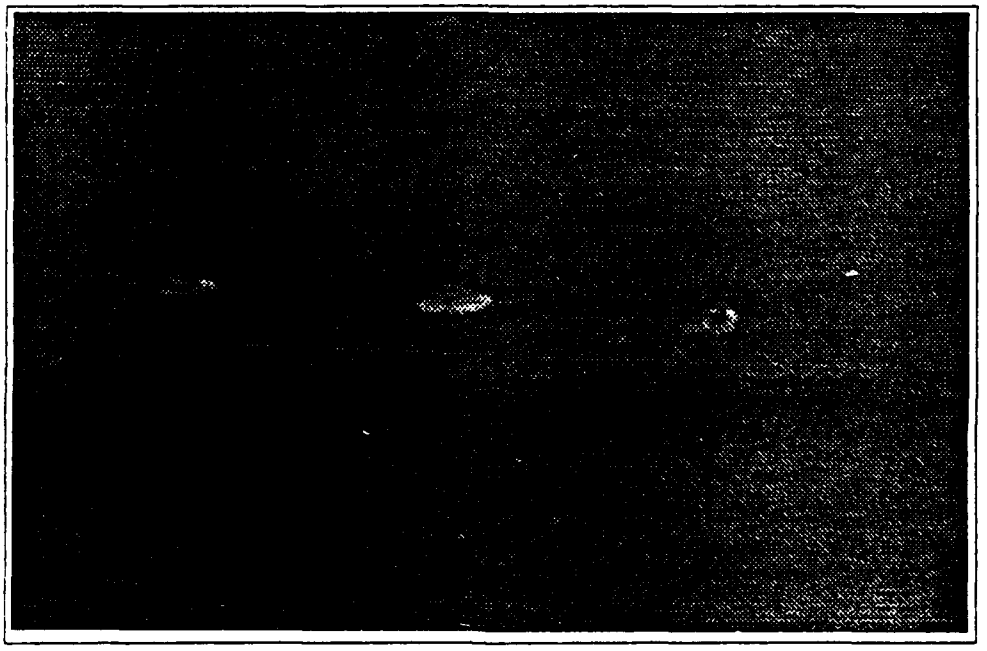


Figure 25. Early-generation IR imagery of an actual T-62, M-60, and a BTR-60. The middle object appears to be the M-60, but this is not completely certain. The object on the far right is the BTR-60.

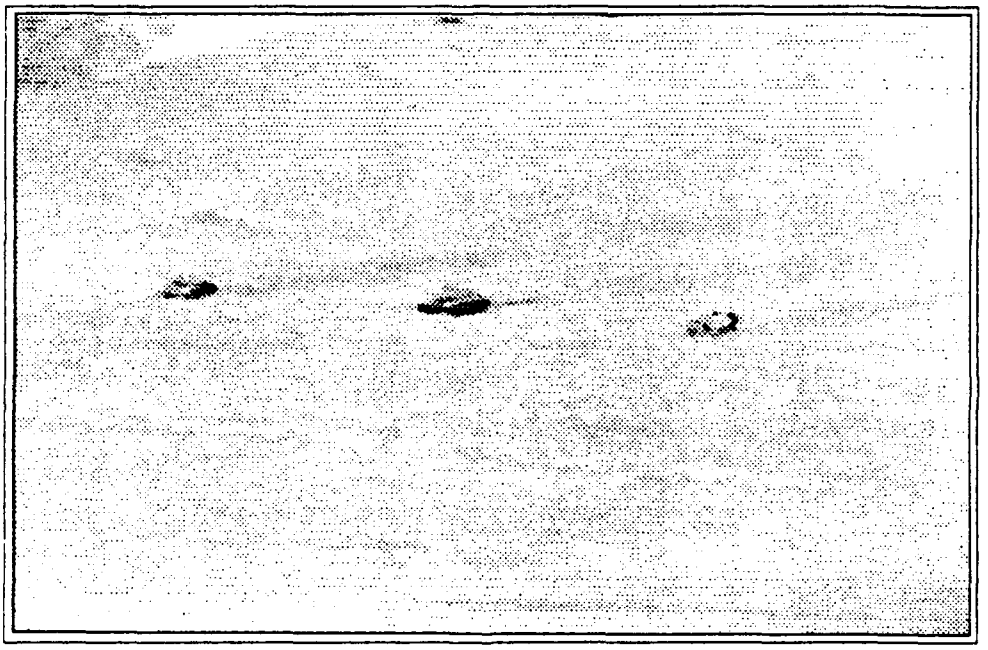


Figure 26. Early-generation IR imagery of an actual T-62, M-60, and a BTR-60. Histogram adjusted to facilitate human viewing and identification of the orientation angles of the objects.

4.5.1.2 Visual Comparisons. Since the two tanks are not readily identifiable, both were treated as candidates for feature extraction and comparison to the synthetic T-62 images. To accomplish this, both candidate objects were visually compared to synthetic images until a best match was found. The orientation angles of the synthetic T-62 were varied until the most-likely orientation angles were obtained. A different set of orientation angles was obtained for each of the two candidate objects.

To account for human errors in matching the synthetic T-62 images to actual imagery, an array of synthetic T-62 images was created for each candidate object (see Figures 27 and 28). In each array, the center image is the synthetic image which appears visually to be the closest match to the candidate object. The features were then extracted from both the actual and the synthetic images. Comparisons of the results appear in the next subsection.

4.5.1.3 Results. The five features were extracted from each candidate image and from each of the 25 synthetic images in the two arrays. However, upon comparing the numerical values of the features extracted from the candidate objects to those of the features extracted from the synthetic objects in the corresponding arrays, the candidates objects did not match any of the synthetic objects (see Tables 5 through to 14). As such, no further analysis was possible.

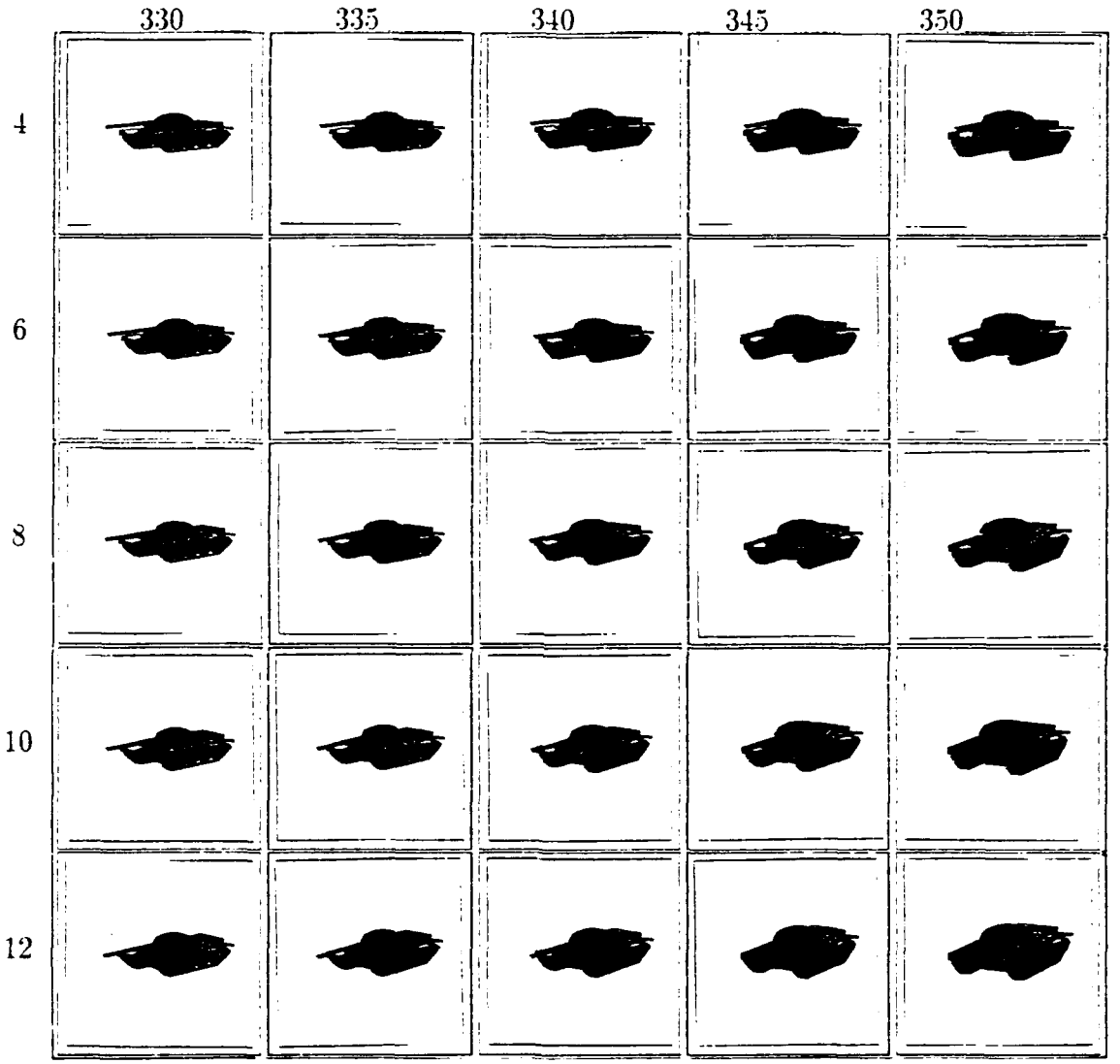


Figure 27. Gray-scale (despite the "binary" look) images for comparison to actual early-generation IR imagery. The center image is the one which was visually the best comparison to the first candidate object (the leftmost object in the early-generation IR imagery—the object which is believed to be the T-62 tank). The numbers across the top represent the sensor view-angle; the numbers down the left side represent the sensor depression-angle.

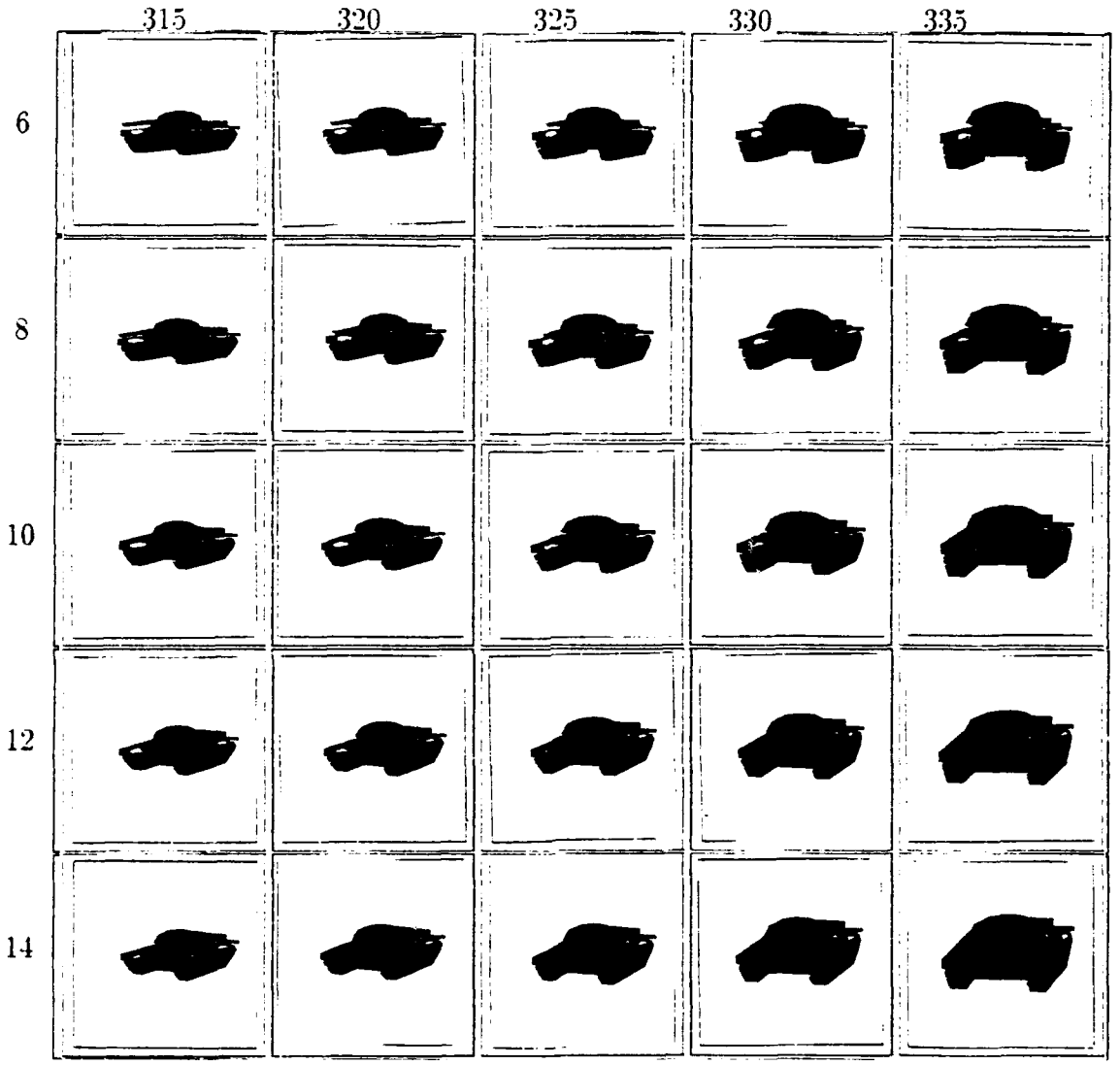


Figure 28. Gray-scale (despite the "binary" look) synthetic T-62 images for comparison to actual early-generation IR imagery. The center image is the one which was visually the best comparison to the second candidate object (the middle object from the early-generation IR imagery—the object which is believed to be the M-60 tank). The numbers across the top represent the sensor view-angle: the numbers down the left side represent the sensor depression-angle.

Table 5. Feature 0 numerical values for the synthetic data. These numbers are for comparison to the values measured from object #1 (the T-62 tank) : 1.000656.

		330 degrees	335 degrees	340 degrees	345 degrees	350 degrees
4		1.001526	1.001576	1.001567	1.001635	1.001657
6		1.001519	1.001529	1.001560	1.001611	1.001655
8		1.001450	1.001458	1.001492	1.001556	1.001603
10		1.001402	1.001426	1.001469	1.001532	1.001579
12		1.001339	1.001388	1.001379	1.001447	1.001475

Table 6. Feature 1 numerical values for the synthetic data. These numbers are for comparison to the values measured from object #1 (the T-62 tank) : 1.000850.

		330 degrees	335 degrees	340 degrees	345 degrees	350 degrees
4		1.001757	1.001847	1.001930	1.002097	1.002225
6		1.001795	1.001860	1.001982	1.002129	1.002275
8		1.001769	1.001850	1.001970	1.002133	1.002308
10		1.001776	1.001880	1.001998	1.002169	1.002353
12		1.001760	1.001900	1.001982	1.002169	1.002373

Table 7. Feature 19 numerical values for the synthetic data. These numbers are for comparison to the values measured from object #1 (the T-62 tank) : 1.001312.

		330 degrees	335 degrees	340 degrees	345 degrees	350 degrees
4		1.002683	1.002821	1.002953	1.003252	1.003409
6		1.002687	1.002786	1.003048	1.003316	1.003507
8		1.002656	1.002926	1.003071	1.003378	1.003685
10		1.002730	1.003001	1.003109	1.003560	1.003759
12		1.002730	1.003050	1.003207	1.003480	1.003900

Table 8. Feature 22 numerical values for the synthetic data. These numbers are for comparison to the values measured from object #1 (the T-62 tank) : 0.989709.

		330 degrees	335 degrees	340 degrees	345 degrees	350 degrees
4		0.990316	0.988583	0.985646	0.983556	0.982220
6		0.987720	0.985759	0.983502	0.981449	0.980563
8		0.986388	0.983309	0.982487	0.980213	0.978453
10		0.984220	0.982161	0.981250	0.978456	0.977061
12		0.983607	0.981075	0.979225	0.977237	0.974678

Table 9. Feature 31 numerical values for the synthetic data. These numbers are for comparison to the values measured from object #1 (the T-62 tank) : 8437624.

		330 degrees	335 degrees	340 degrees	345 degrees	350 degrees
4		8376295	8369374	8352880	8338793	8315377
6		8375866	8364977	8351955	8334781	8312120
8		8375777	8362082	8346704	8328955	8306396
10		8372513	8357977	8342978	8324641	8300182
12		8369807	8354149	8337926	8317819	8290251

Table 10. Feature 0 numerical values for the synthetic data. These numbers are for comparison to the value measured from object #2 (the M-60 tank) : 1.000931.

		315 degrees	320 degrees	325 degrees	330 degrees	335 degrees
6		1.001542	1.001545	1.001493	1.001519	1.001529
8		1.001533	1.001494	1.001449	1.001450	1.001458
10		1.001466	1.001449	1.001426	1.001402	1.001426
12		1.001443	1.001373	1.001332	1.001339	1.001388
14		1.001392	1.001334	1.001295	1.001300	1.001312

Table 11. Feature 1 numerical values for the synthetic data. These numbers are for comparison to the value measured from object #2 (the M-60 tank) : 1.001192.

		315 degrees	320 degrees	325 degrees	330 degrees	335 degrees
6		1.001697	1.001727	1.001711	1.001795	1.001860
8		1.001719	1.001711	1.001716	1.001769	1.001850
10		1.001690	1.001720	1.001749	1.001776	1.001880
12		1.001706	1.001699	1.001723	1.001760	1.001900
14		1.001705	1.001702	1.001717	1.001797	1.001891

Table 12. Feature 19 numerical values for the synthetic data. These numbers are for comparison to the value measured from object #2 (the M-60 tank) : 1.001832.

		315 degrees	320 degrees	325 degrees	330 degrees	335 degrees
6		1.002662	1.002607	1.002575	1.002687	1.002786
8		1.002761	1.002568	1.002595	1.002656	1.002926
10		1.002684	1.002669	1.002693	1.002730	1.003001
12		1.002806	1.002687	1.002738	1.002730	1.003050
14		1.002858	1.002766	1.002816	1.002863	1.003117

Table 13. Feature 22 numerical values for the synthetic data. These numbers are for comparison to the value measured from object #2 (the M-60 tank) : 0.989153.

		315 degrees	320 degrees	325 degrees	330 degrees	335 degrees
6		0.992820	0.991455	0.990017	0.987720	0.985759
8		0.990870	0.989665	0.987964	0.986388	0.983309
10		0.989428	0.987546	0.985515	0.984220	0.982161
12		0.987504	0.985899	0.983802	0.983607	0.981075
14		0.986709	0.984742	0.982698	0.981264	0.979540

Table 14. Feature 31 numerical values for the synthetic data. These numbers are for comparison to the value measured from object #2 (the M-60 tank) : 8395579.

		315 degrees	320 degrees	325 degrees	330 degrees	335 degrees
6		8392660	8390946	8386028	8375866	8364977
8		8391338	8389001	8383864	8375777	8362082
10		8390164	8385628	8381103	8372513	8357977
12		8388543	8384341	8373904	8369807	8354149
14		8385905	8379978	8373171	8362162	8348651

As can be seen from the tables of data, except for feature 22, all of the actual measured values were out of the range of the values in the tables (for both objects). For object #1 (the candidate T-62 tank), all the numerical values tend to suggest smaller sensor view and depression angles; however, visually comparing the object to the imagery in Figure 27 precludes this as an option. For object #2 (the candidate M-60 tank), the numerical data does not suggest any one specific change (feature 22 does suggest smaller sensor view and depression angles, but, as with the other object, visual comparisons to the imagery in Figure 28 preclude this as an option).

4.5.1.4 Discussion. Possible reasons for the poor comparisons between the synthetic IR imagery and the actual early-generation IR imagery are

- Too few pixels on the target. The objects in the actual IR imagery had fewer pixels on the target than were provided by the synthetic imagery. Object #1 had 277 pixels on the target; the corresponding synthetic image had between 513 and 962 pixels on the target. Object #2 had 402 pixels on the target; the corresponding synthetic images had between 437 and 624 pixels on the target. This discrepancy is due both to the resolution of the actual IR imagery (496 across by 320 down) compared to the synthetic imagery resolution (512 by 512) and to the range at which the actual imagery was taken (2010 meters). This might have been less of a problem had the range for the actual imagery been shorter (giving more pixels on the target and a corresponding lessening of the impact of minor shape variations).

  Note: the number of pixels on the target for the actual IR imagery is less than required in some ATR specifications (34). In these specifications, at least 432 pixels are required on a target which is at a distance of approximately 1 km.

- Poor "truthing" of the actual IR imagery. As can be seen from Figure 29, the actual IR imagery had amorphous blobs representing the objects in the imagery. The poor quality of these blobs directly affects numbers calculated from their shapes.

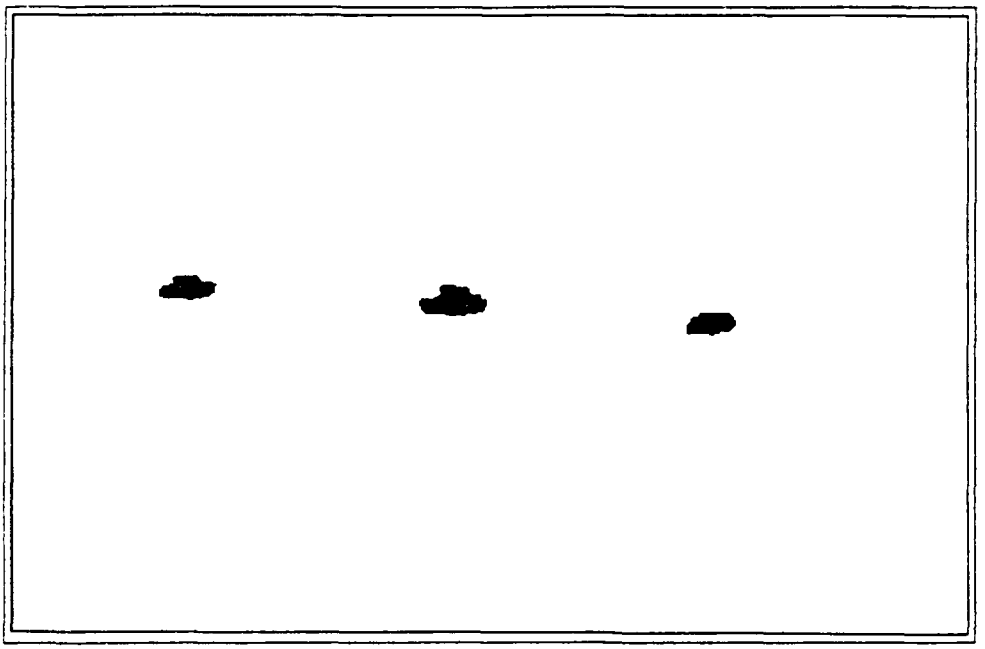
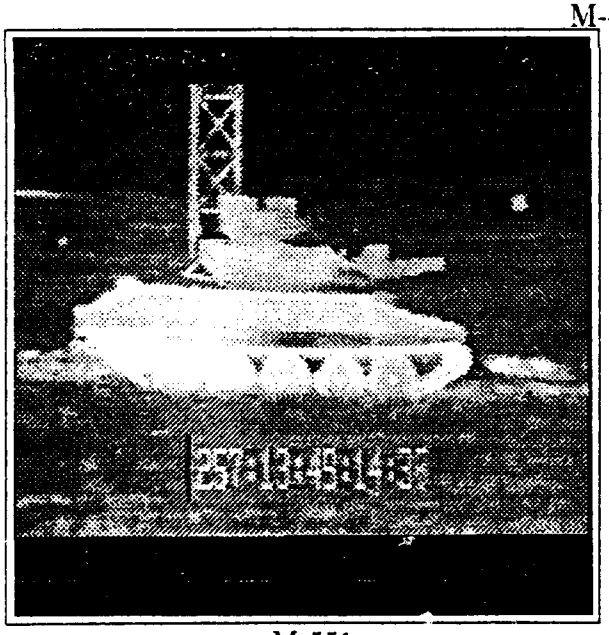


Figure 29. "Truthed" data for early-generation IR imagery of an actual T-62, M-60, and a BTR-60. The middle object appears to be the M-60, but this is not completely certain. The object on the far right is the BTR-60.

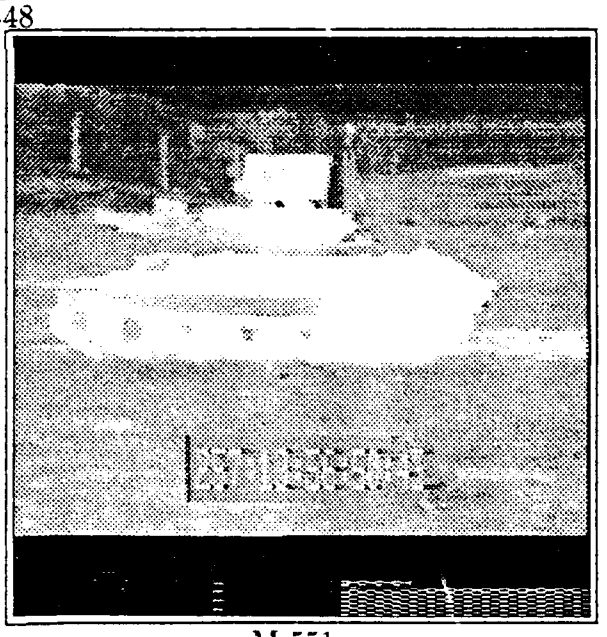
- Poor selection of features. Though the features selected showed good results for synthetic object orientation determination, the features may not be realistic for real-world applications.

4.5.2 Later-Generation IR-Sensor Imagery. This section also compares the values measured from the synthetic T-62 image to objects in actual IR imagery. In this section, the imagery was obtained using later-generation IR sensor. The images are of a U.S. M-551 infantry fighting vehicle (see Figure 30) and an M-48 main battle tank. These were used since no later-generation imagery of a T-62 tank was obtainable. These images were selected because they were taken at zero-degree sensor depression-angle and represent sensor view-angles of 45, 90, and 270 degrees. At these angles, the best determination of whether these images can be compared to the T-62 imagery should be possible.

M-48



M-551



M-551

Figure 30. Later-generation IR imagery.

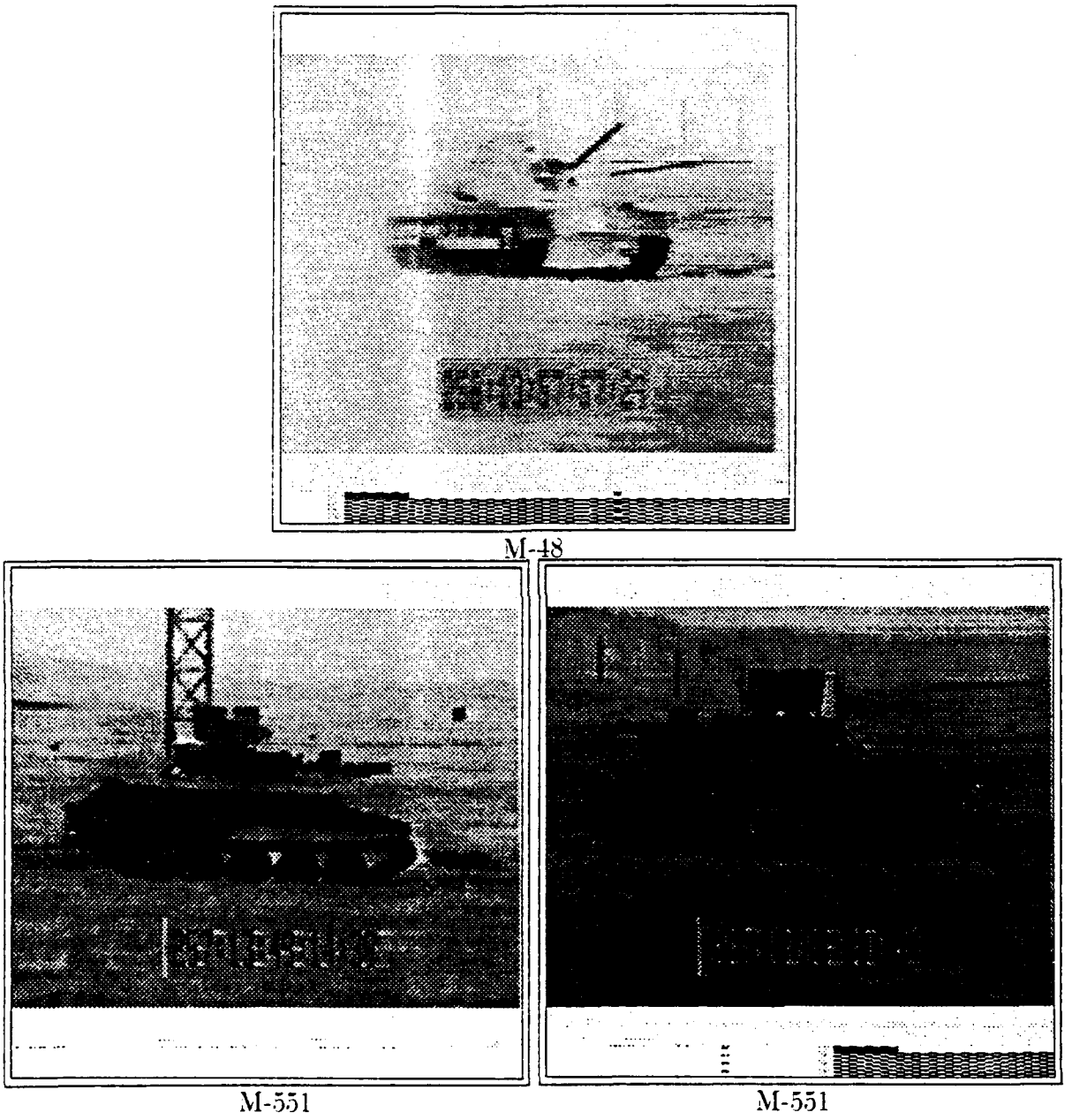
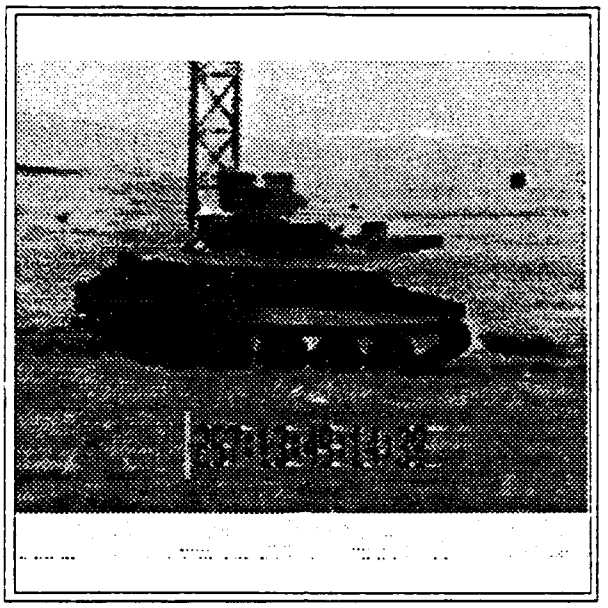
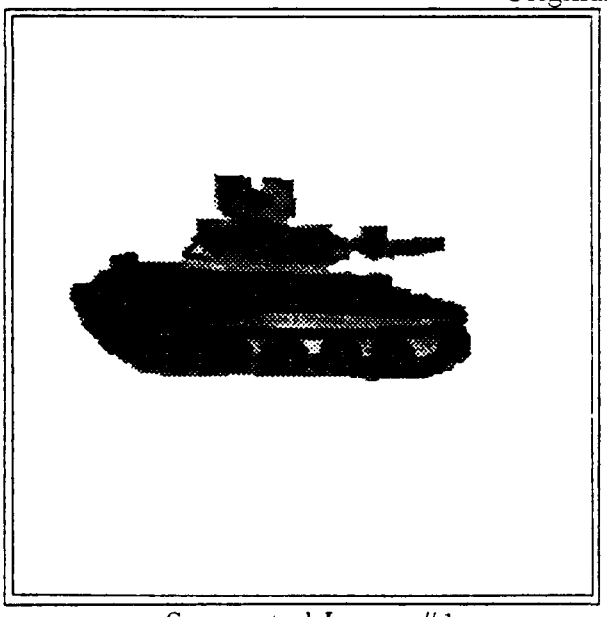


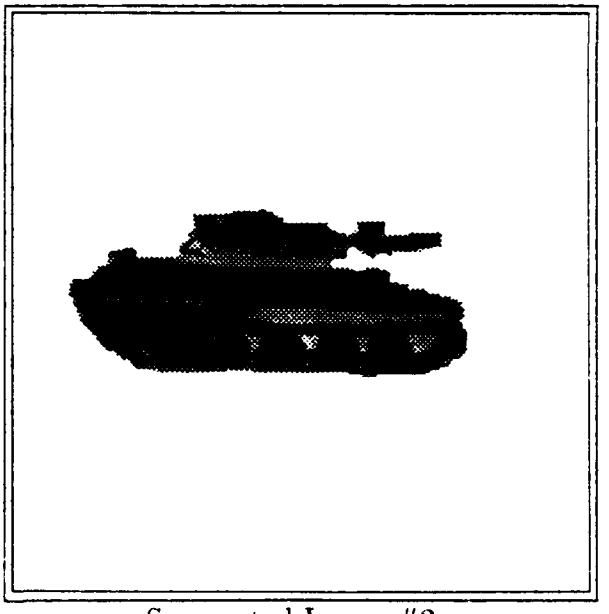
Figure 31. Later-generation IR imagery with the intensity reversed (essentially, the "negatives" of the actual imagery).



Original M-551

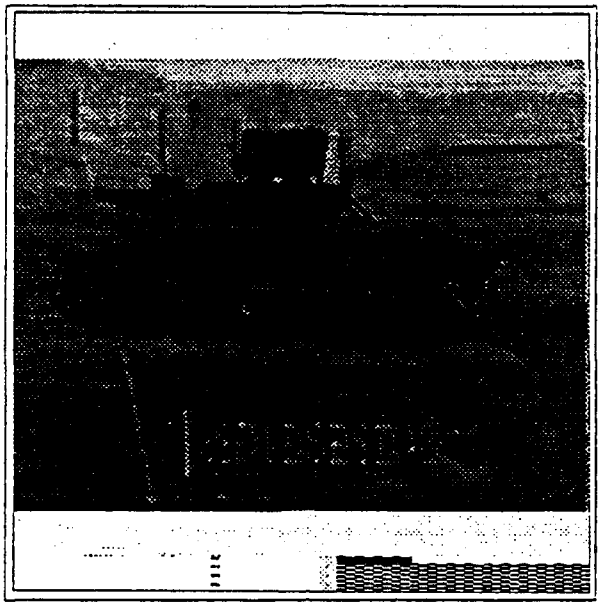


Segmented Image #1

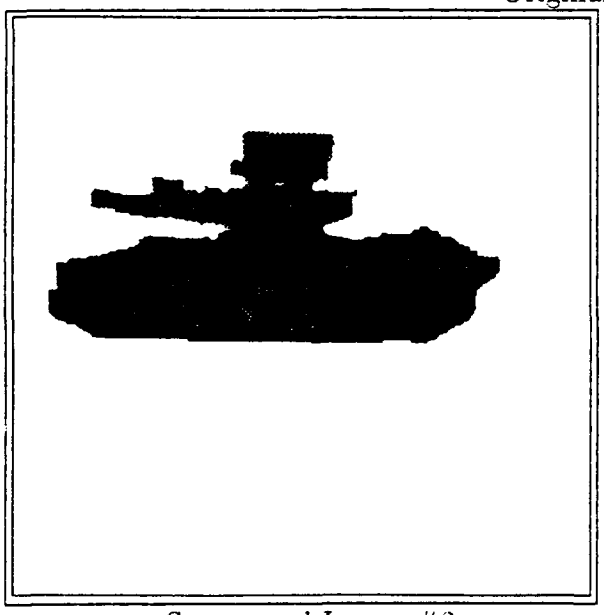


Segmented Image #2

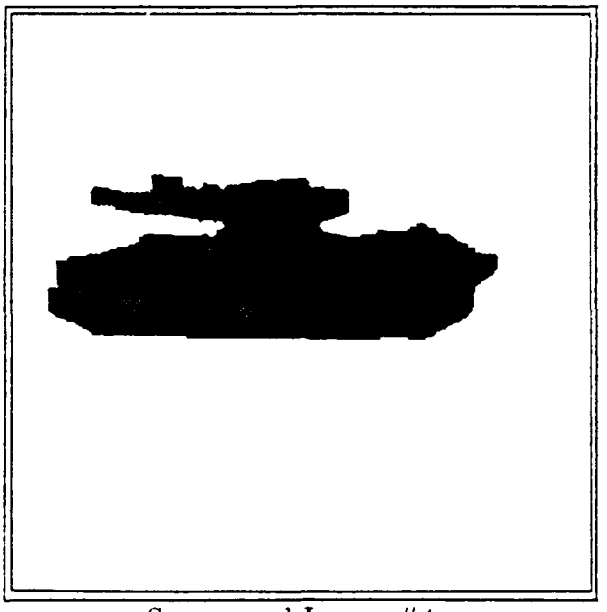
Figure 32. Hand segmentation used on later-generation IR imagery.



Original M-551



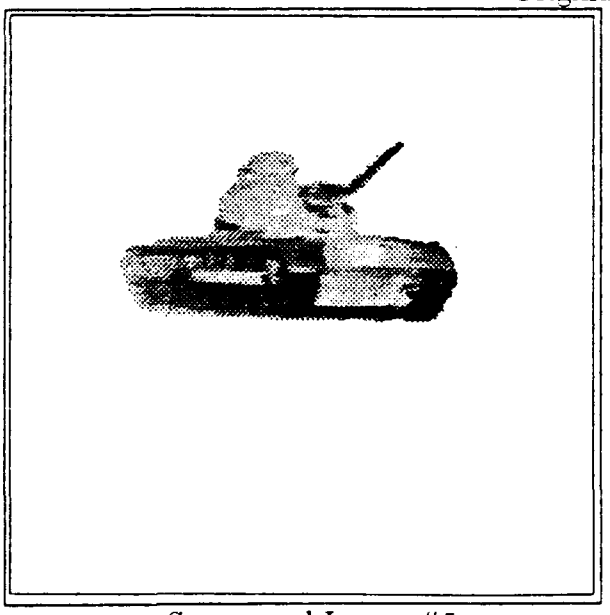
Segmented Image #3



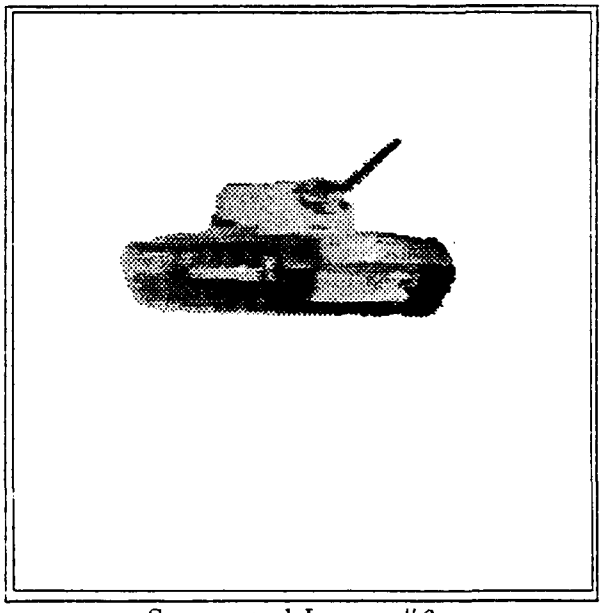
Segmented Image #4

Figure 33. Hand segmentation used on later-generation IR imagery.

Original M-48



Segmented Image #5



Segmented Image #6

Figure 34. Hand segmentation used on later-generation IR imagery.

Table 15. Results of feature extraction on later-generation IR imagery.

Image		Feature 0	Feature 1	Feature 19	Feature 22	Feature 31
1		1.081948	1.106509	1.165127	0.884521	7764919
2		1.095708	1.114226	1.184023	0.890950	7779472
3		1.102334	1.135588	1.188428	0.897199	7769165
4		1.124553	1.144689	1.199172	0.923797	7782251
5		1.056558	1.077580	1.129388	0.888165	7825298
6		1.063974	1.082536	1.135558	0.890500	7838402

To facilitate human viewing of the imagery, the imagery has been intensity-reversed (i.e., the "negative" of the imagery is provided) (see Figure 31). For the feature extraction, the images were hand segmented and numbered as shown in Figures 32, 33 and 34. The segmentation is coarse to simulate actual computer segmentation results. For the each image, two segmented images were created: one with the cupola (M-48) or turret-mounted machine-gun (M-551) in place; the other, without any objects atop the turret. As a result, there are six segmented images for processing.

4.5.2.1 Results. The numerical results are tabulated (Table 15) and plotted upon the T-62 data (see Figures 35 to 39). In the figures, the boxes show where the values should have been for the specific image being evaluated. As can be seen from the location of the boxes for each of the data points, the later-generation imagery does not consistently correspond to that for the synthetic T-62 imagery for each of the five selected features. For some of the imagery, a particular feature may have yielded a value close to that of the correspondingly oriented synthetic imagery; however, no consistency can be found when accounting for all five features.

NOTE: the image number (1 through 6) is for the hand-segmented images as numbered in Figures 32, 33, and 34.

4.5.2.2 Discussion. Possible reasons for the poor comparisons between the synthetic IR imagery and the actual later-generation IR imagery are

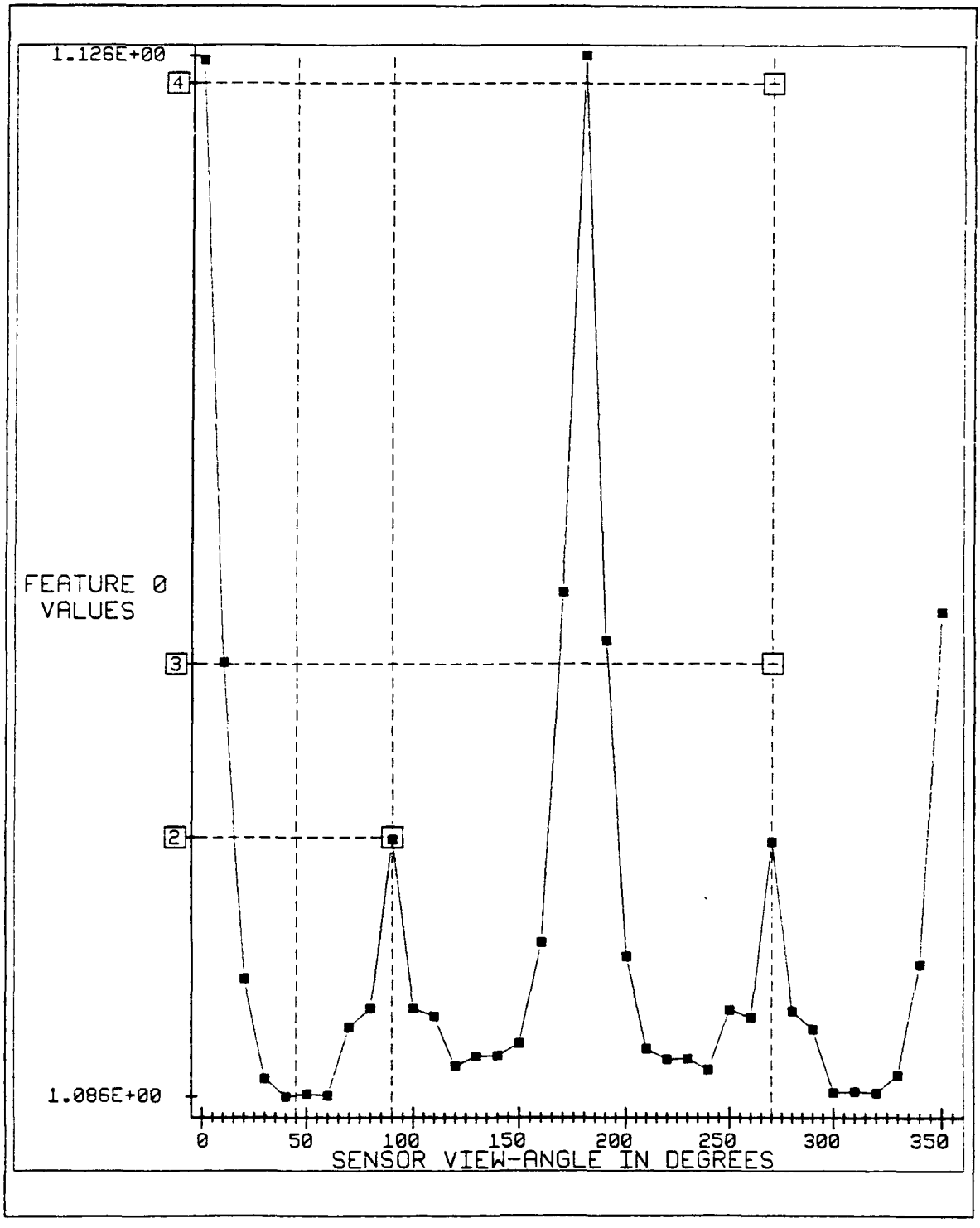


Figure 35. Later-generation IR imagery features plotted against synthetic imagery's corresponding feature values (Feature 0—silhouette moments: M_{20}/M_{02}). The synthetic data was generated at a 0-degree sensor depression-angle.

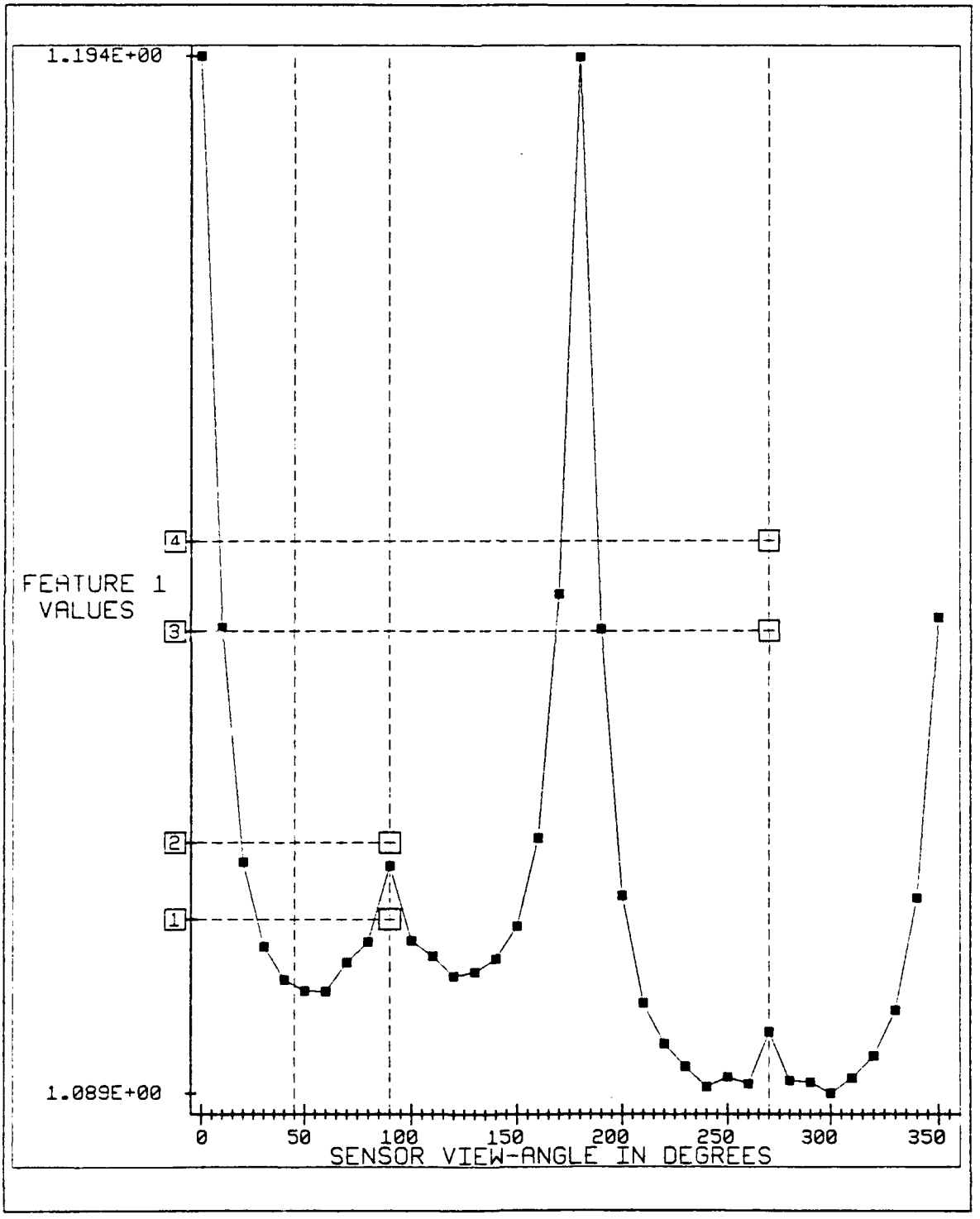


Figure 36. Later-generation IR imagery features plotted against synthetic imagery's corresponding feature values (Feature 1—silhouette moments: M_{20}/M_{11}). The synthetic data was generated at a 0-degree sensor depression-angle.

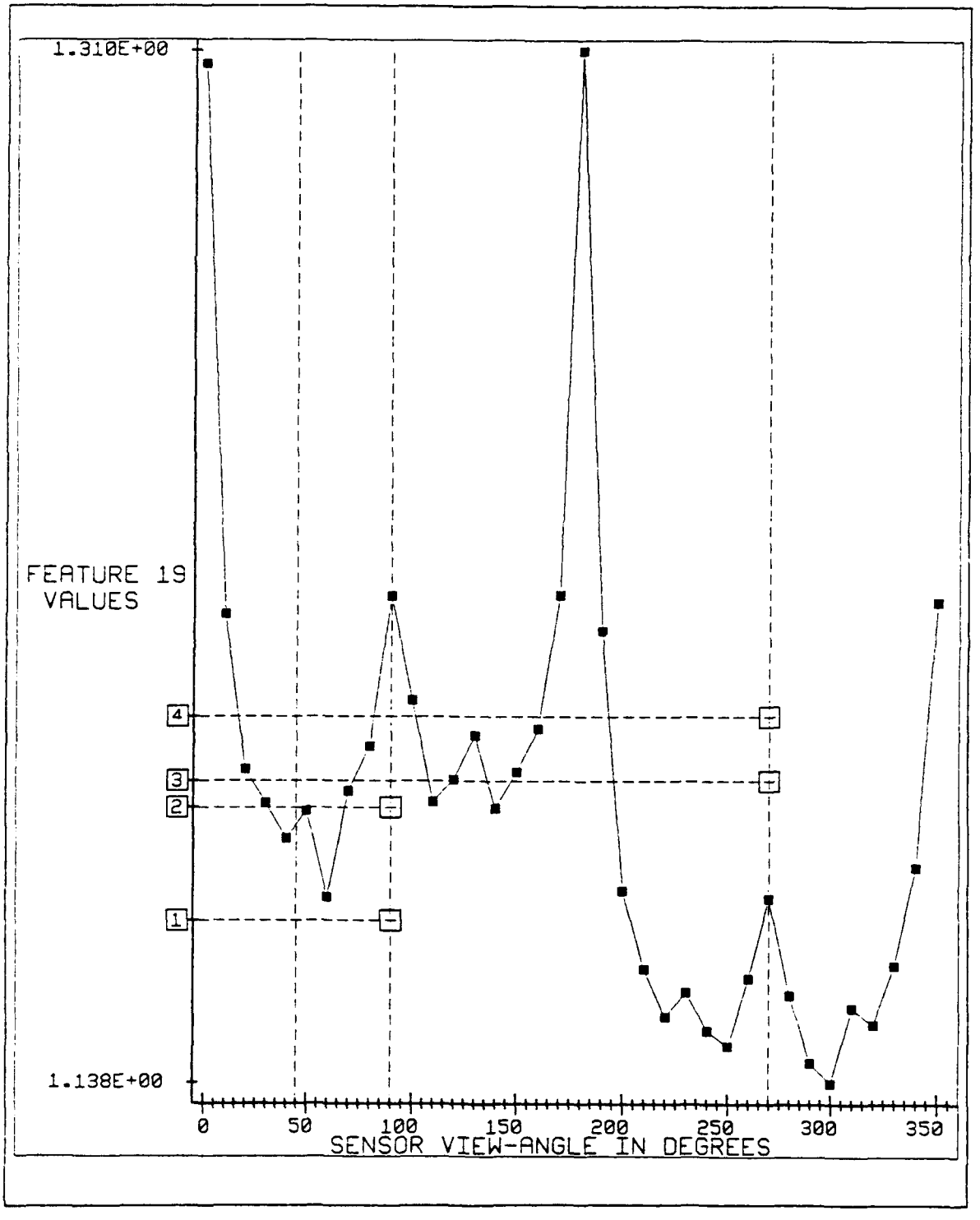


Figure 37. Later-generation IR imagery features plotted against synthetic imagery's corresponding feature values (Feature 19—outline moments: M_{20}/M_{11}). The synthetic data was generated at a 0-degree sensor depression-angle.

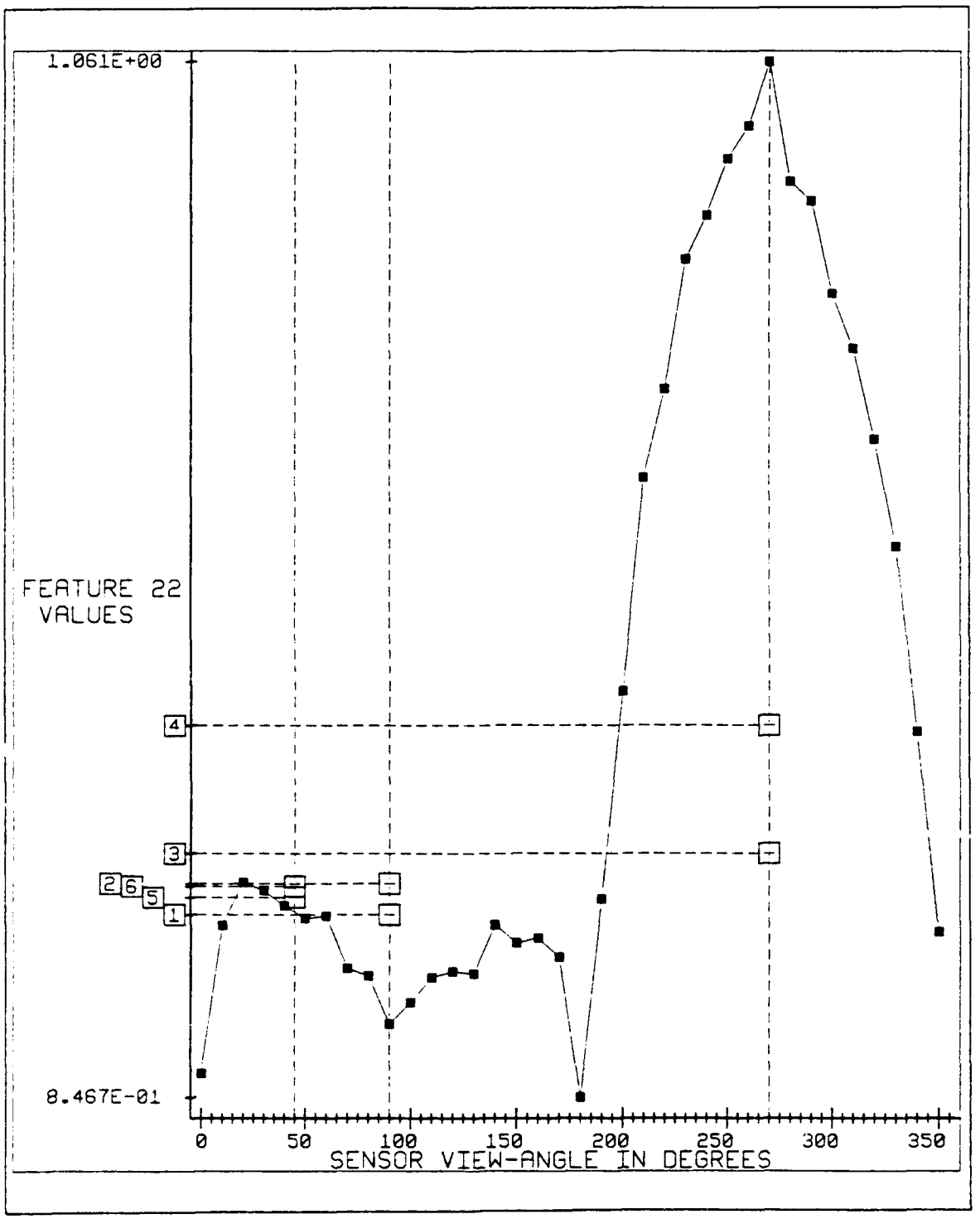


Figure 38. Later-generation IR imagery features plotted against synthetic imagery's corresponding feature values (Feature 22—outline moments: M_{30}/M_{21}). The synthetic data was generated at a 0-degree sensor depression-angle.

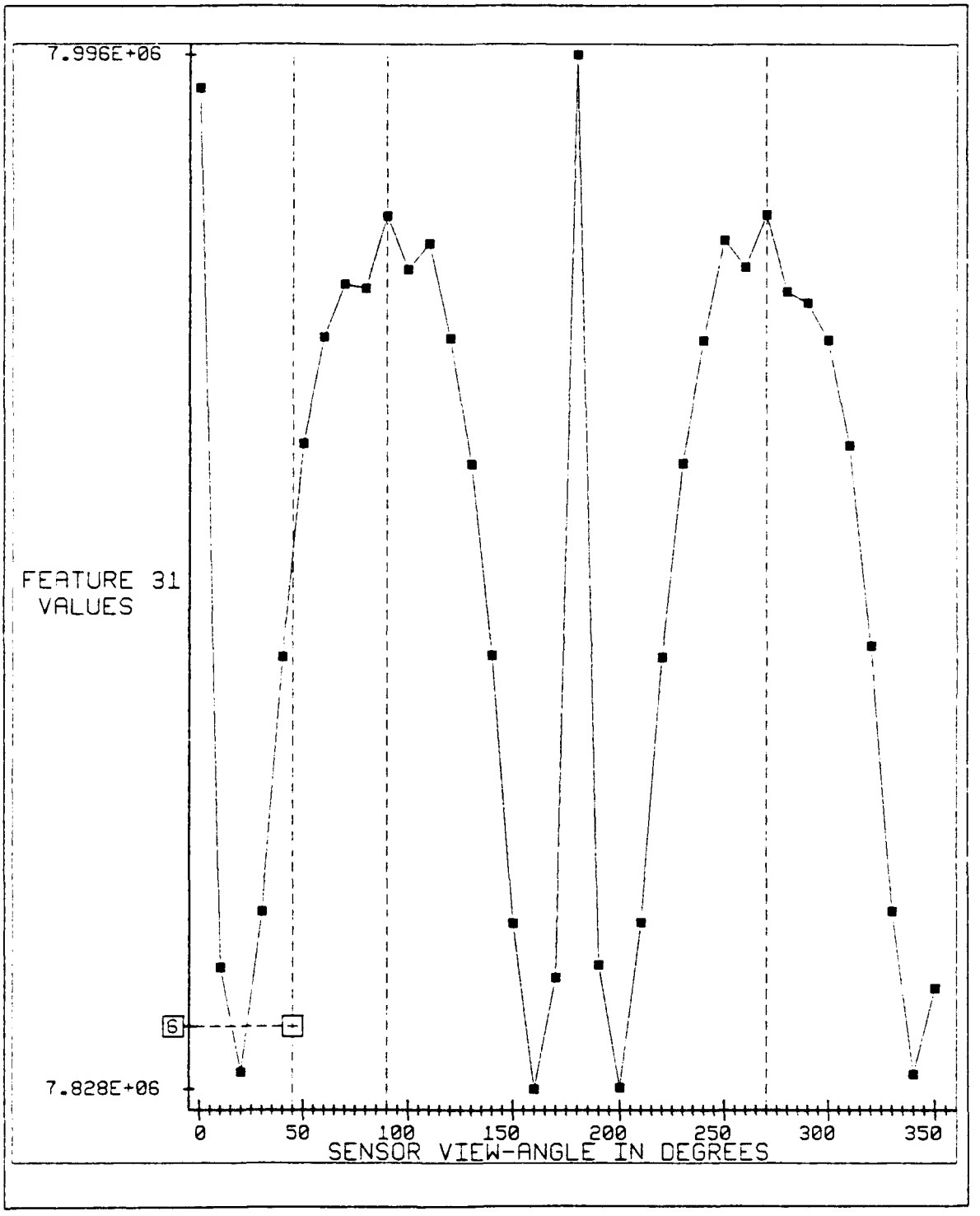


Figure 39. Later-generation IR imagery features plotted against synthetic imagery's corresponding feature values (Feature 31—outline moments: $M_{03}/(M_{00})^2$). The synthetic data was generated at a 0-degree sensor depression-angle.

- The objects are not comparable. Comparing a Soviet T-62 tank to a U.S. M-48 tank is a reasonably fair comparison; however, it is still a comparison of "apples and oranges." The comparison between the M-551 imagery and the synthetic T-62 imagery is especially a mismatched comparison (size of turret and length of barrel being two most notable differences between the T-62 and the M-551). As an extension of this thought, though, these five shape features may be able to discriminate T-62's from M-48's, M-551's, and other non-T-62 objects.
- The objects in the actual IR imagery are too poorly segmented for the feature calculations. Portions of the objects may have been erroneously removed, and, furthermore, portions of the background may have been erroneously included. Though it should be noted that this is a problem that all segmentation systems would probably exhibit.
- The zero-degree sensor depression-angle creates too much error between the synthetic and any real imagery. As was seen earlier, at this sensor depression-angle, the bottom of the tread is not present in the synthetic imagery. As such, the jagged and squarish quality of the bottom of the T-62 imagery might yield incomparable results even to actual later-generation IR imagery of a T-62.
- Poor selection of features. As is possible with the early- generation IR imagery, these features may not be extendable from synthetic IR imagery to real IR imagery.

4.6 Summary

In summary, by using the five selected features and the technique of sorting the imagery into consecutively smaller angle-bins, the T-62 tank within the synthetic IR imagery was sorted into the correct 15-degree angle-bin (for the sensor view-angle) with an accuracy better than 90%. However, when the features were extracted from real IR imagery and compared to those features extracted from the synthetic imagery of a T-62 tank, no conclusive results were obtained. As such, the applicability of the selected features and the technique for real-world IR imagery could not be ascertained.

V. Conclusions and Recommendations

5.1 Conclusions

For an accurate comparison of features and techniques for pose estimation using computer models, the modeling program must create synthetic imagery mathematically comparable to real-world imagery. As seen in this research, the rendering software had minor defects which made it too mathematically and, in some instances, even too visually inexact for a complete investigation into the problem addressed: finding those features and techniques which may be extended from synthetic to real-world IR imagery for pose estimation.

As such, no conclusions can be drawn on the use of silhouette and outline shape moments as features for the aforementioned comparison. However, given only the requirement to determine the base-plane rotation angle of the object within the synthetic imagery, these moments can be used (up through the second-order moments). With better than 90% accuracy, the base-plane rotation of the synthetic object can be determined to within $\pm$ 7.5 degrees.

Key to this accuracy with the synthetic imagery is the use of the hierarchical sorting technique whereby the object's rotation is first coarsely determined (e.g., the viewing quadrant or 90-degree angle-bin is determined) and then finer determinations are made (e.g., determination of the 45-degree angle-bin within the 90-degree angle-bin followed by determination of the 15-degree angle-bin within the 45 degree angle-bin). This hierarchical techniques reduces the impact of the ambiguities associated with the use of second-order moments.

Lastly, to perform the hierarchical sorting, a relatively small neural network configured with only two hidden layers (ten nodes in the first layer; five in the second) can be used. Though any training paradigm may have been equally successful, "backpropagation with momentum" was shown to be more than adequate for the task.

5.2 Recommendations

There are two basic recommendations based on this research. The first recommendation involves assessing the many modeling programs and comparing the imagery to real-world imagery. This comparison has to be more rigorous than merely requiring the model appear visually accurate at long ranges—the model must appear mathematically accurate at all realistic ranges. At a very minimum, comparable values should be obtained from extracting the shape moments from the synthetic and the real-world imagery (especially since these scalar features are not based on IR modeling as much as geometric modeling). Implied in this recommendation is the requirement for close-up IR imagery of whatever object the image-modeling/rendering software can produce.

The second recommendation involves the use of the "second choice" features and techniques identified in Chapter 2. The features were hotspot centroid and two-dimensional aspect ratios (height-to-width). In using these features, some metric would be used to locate the position of the hotspot relative to the overall shape of the object (see Figure 40 for an example of the hotspot location for one orientation of the synthetic T-62 tank—the program to create this imagery is available through the thesis advisor for this research).

For example, as shown in Figure 41, for a given value for the aspect ratio, there are many object rotations which could have yielded that specific aspect ratio. However, as shown in Figure 42, when the location of the hotspot centroid is found for six images having the same aspect ratio, the location of the hotspot centroid appears unique. Using a metric such as shown in Figure 43, it may be possible to determine both the sensor view-angle as well as the sensor depression angle.

However, there are some pitfalls to be aware of with this technique. Most notably, the hotspot may not always be centered about the engine area of the object. For example, when using this technique for the candidate T-62 tank in Figure 25, and the closest matching synthetic T-62 in Figure 44, the turret occludes part of the hotspot. Therefore, for comparison to the synthetic imagery, the synthetic imagery must accurately reflect the heat

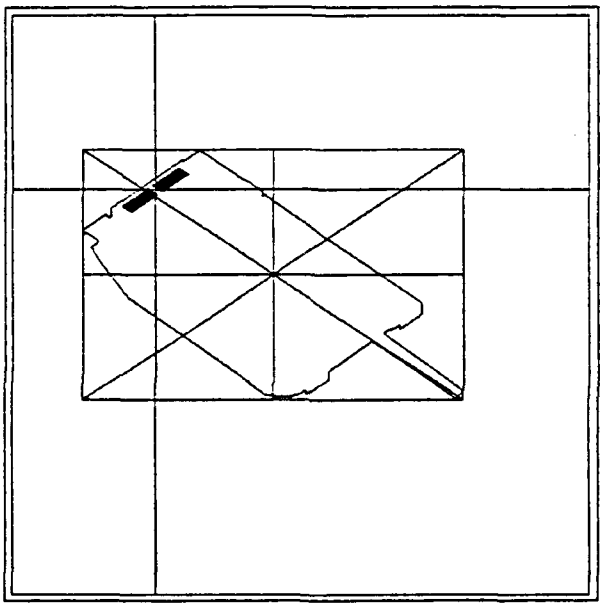


Figure 40. Example of the location of hotspot centroid for a synthetic T-62 tank. The large crosshairs identify the centroid of the hotspots (which are shown as black squares). The outline of the T-62 and the box about its outline are provided for reference.

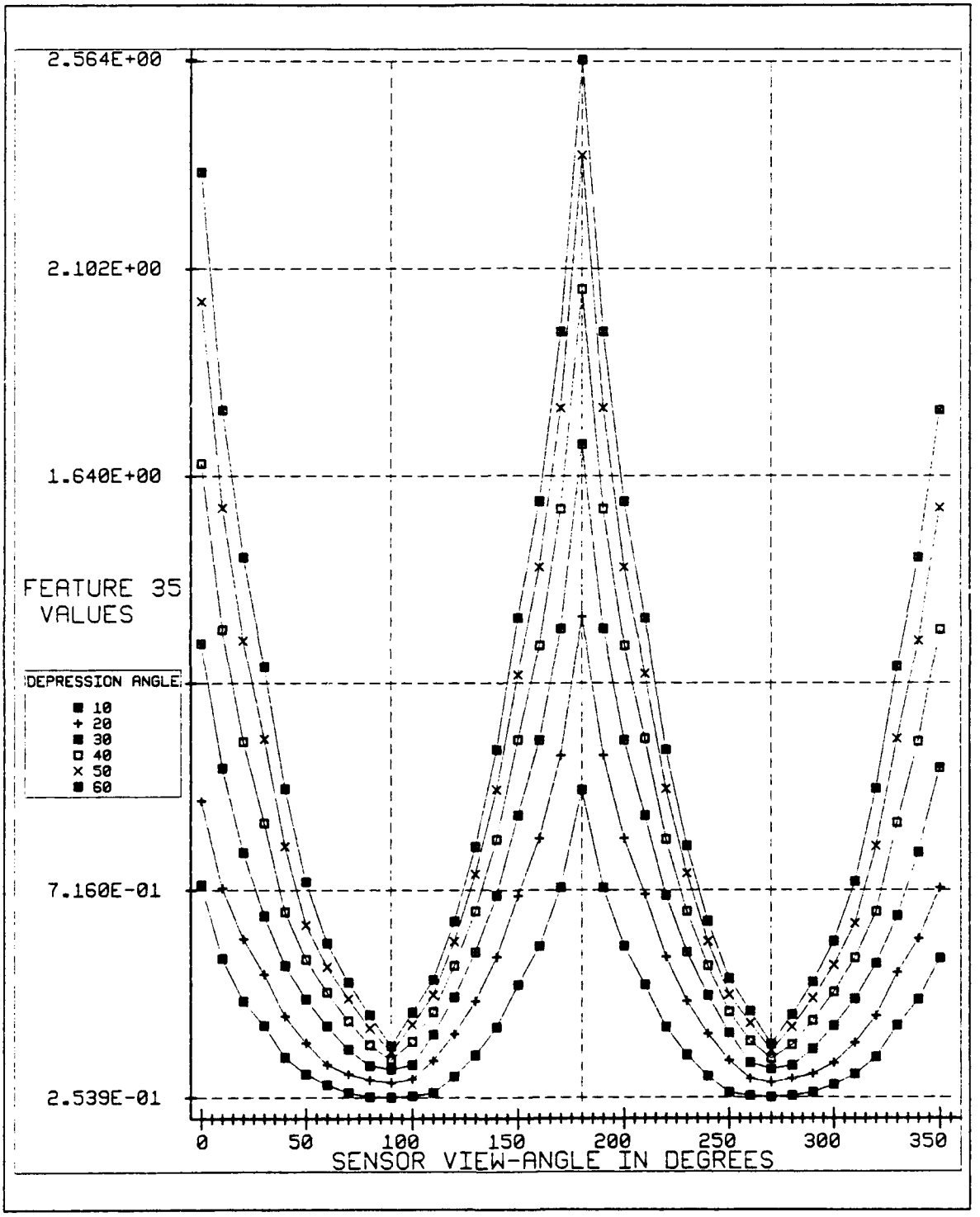


Figure 41. Feature 35 (aspect ratio) plotted for the six sensor depression-angles and the 36 sensor view-angles.

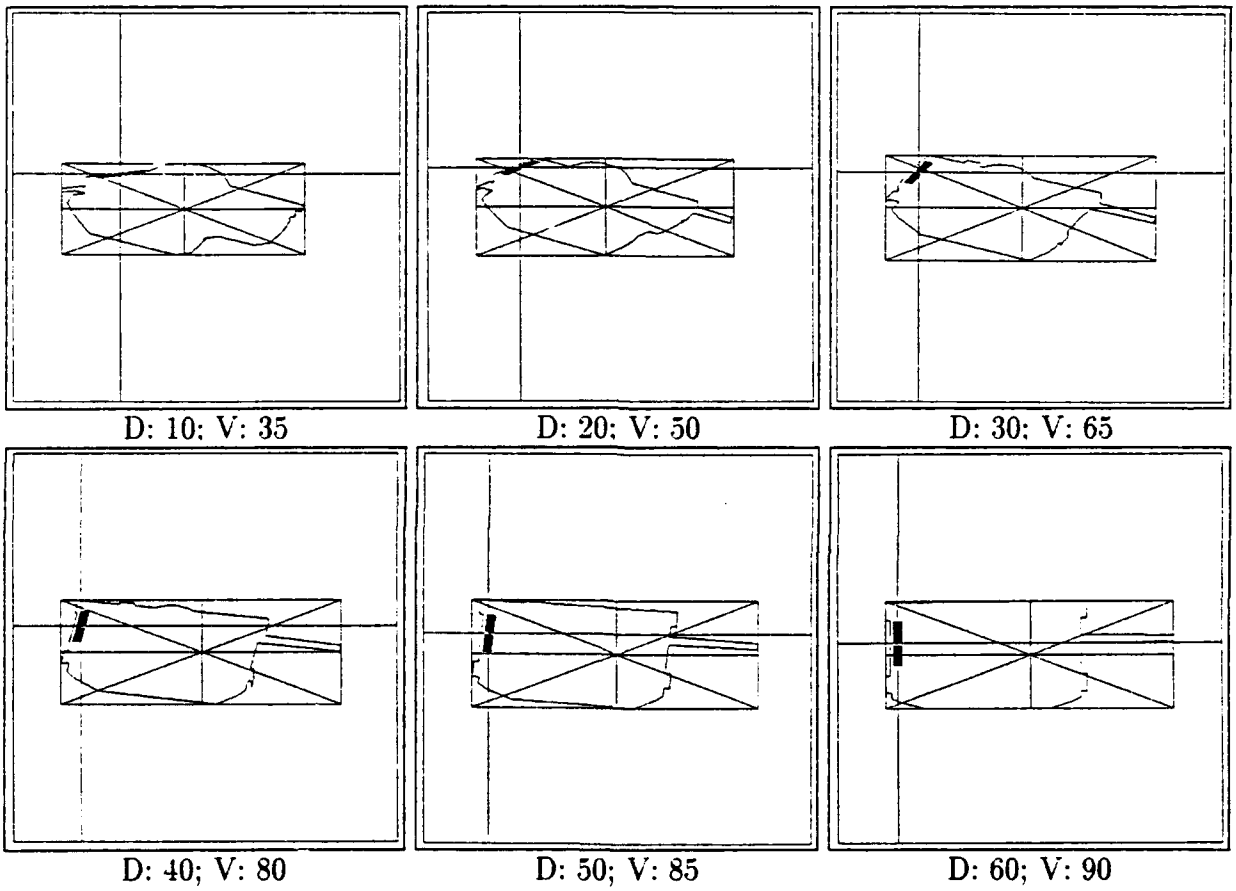


Figure 42. Locations of the hotspot centroids for six images having the same aspect ratios. For each image, the sensor depression-angle (D) and sensor view-angle (V) are given. Note the location of the hotspot appears to be unique for each of these images.

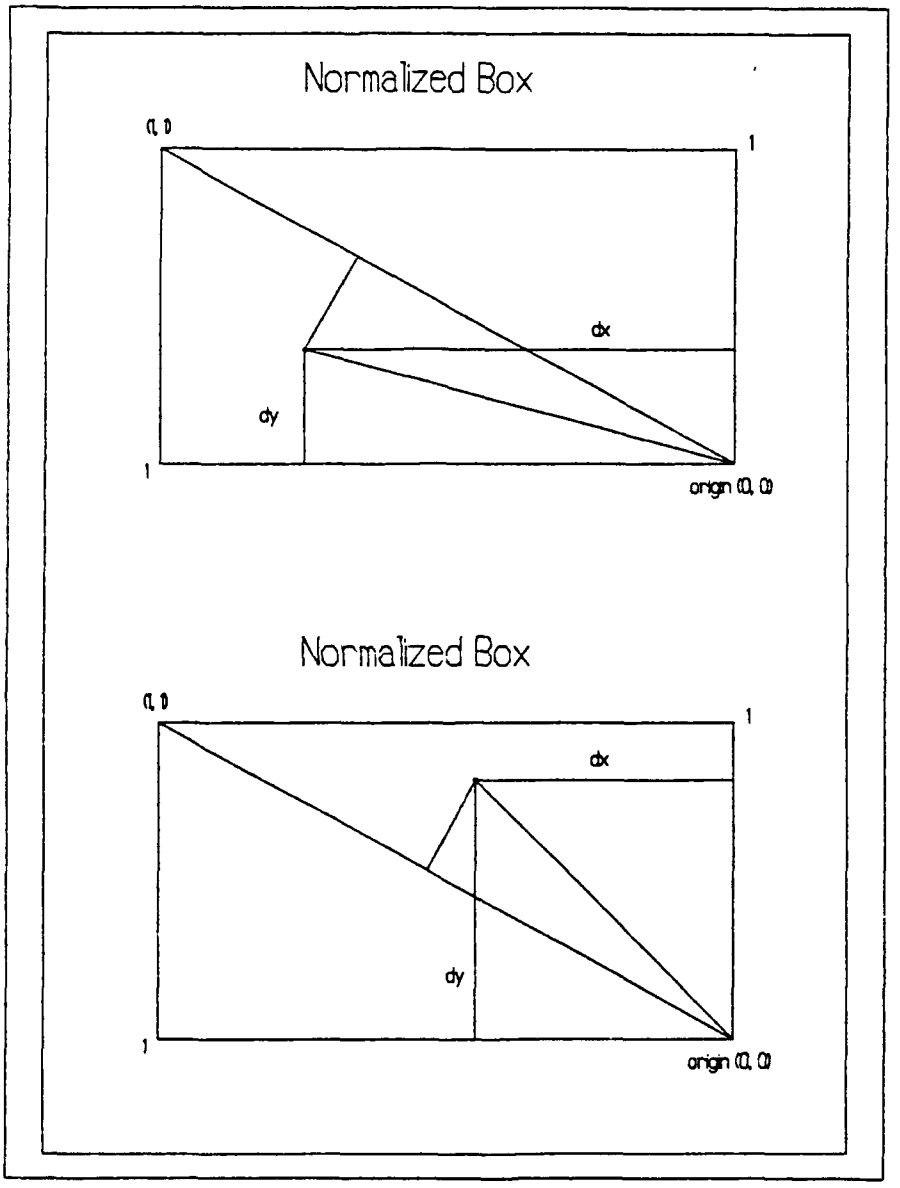


Figure 43. Potential hotspot metric for determining relative location of the hotspot centroid within a normalized image.

conduction to other parts of the tank; otherwise, the hotspot centroid of the real-world T-62 will be located more toward the upper right corner of the object. In fact, in comparing the locations of the hotspots for the two images, as expected, the synthetic imagery's hotspot is not comparable to the real-world imagery's hotspot (see Figure 45 for relative locations of the hotspot centroids for the two images).

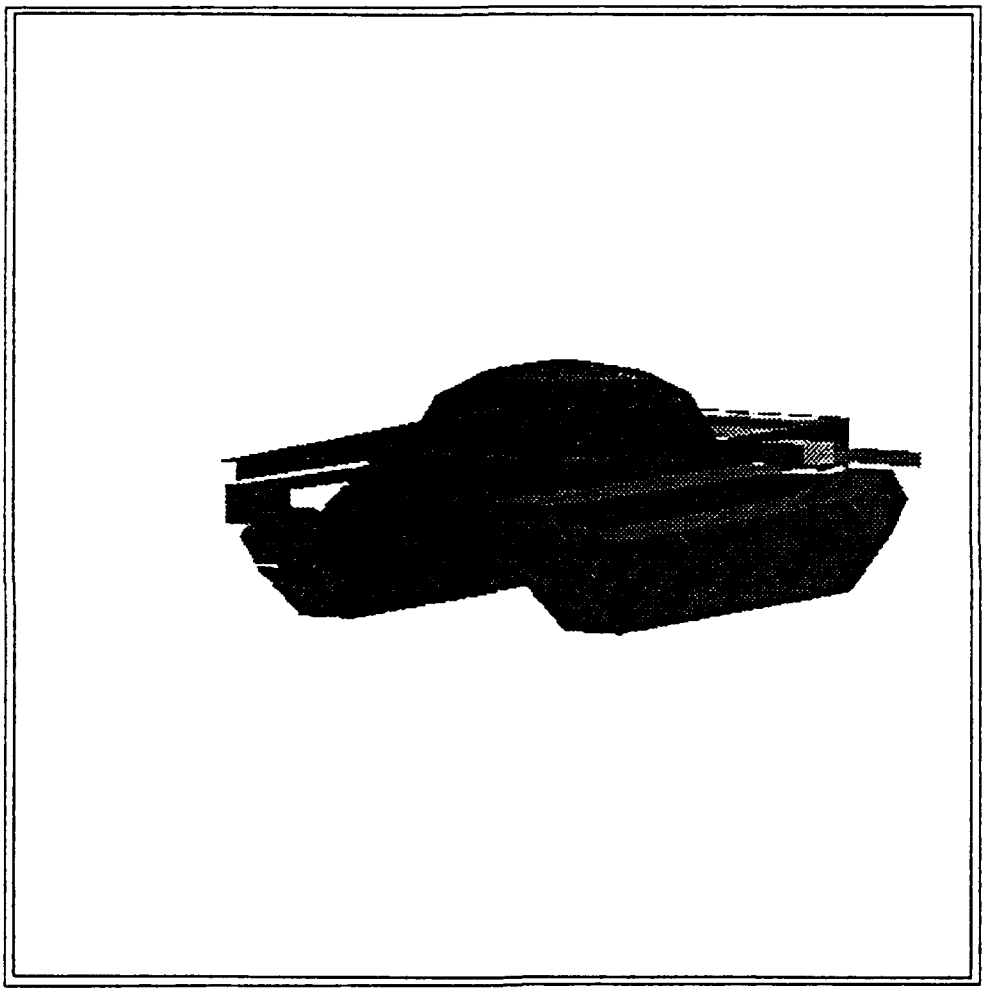


Figure 44. Synthetic imagery showing occlusion of the hotspot by the turret. Note that the hotspot (at the rear of the tank) has not been conducted to the extremes of the tank (even though the tank has been "idling" in the simulation for six hours).

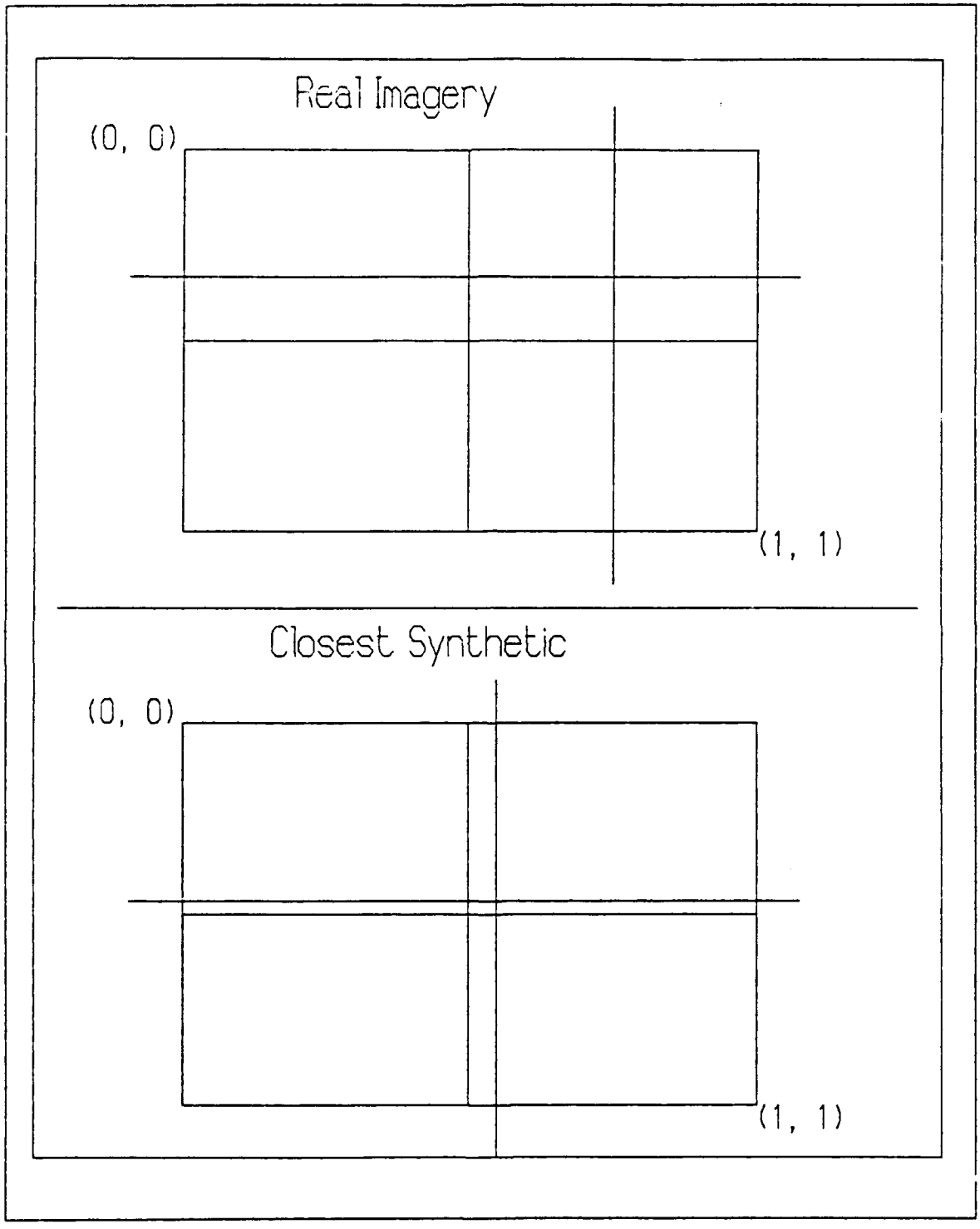


Figure 45. Comparison of the location of the centroid of the hotspot between the normalized images of the real-world T-62 and the closest matching (both in aspect ratio and in hotspot location) synthetic T-62.

Appendix A. *Neural Graphics Conventions*

A.1 Neural Graphics Input Data

For input, Neural Graphics uses the data file specified in **setup.fil**. For the **setup.fil** shown above in Section 3.3.4.1, **data.file** would contain the training and test inputs for Neural Graphics. This file was of the format (using an arbitrary example for purposes of illustration):

```
5 6 2 3
1  1.0 2.0  1
2  1.1 2.1  1
3  1.2 2.2  2
4  1.3 2.3  3
5  1.4 2.4  3
6  1.5 2.5  1
7  1.6 2.6  1
8  1.7 2.7  2
9  1.8 2.8  2
10 1.9 2.9  3
11 1.9 2.9  3
```

The first line within this file indicates the configuration of the training and test data. For the example above, there are five (5) training vectors and six (6) test vectors. The training and test vectors each have two (2) elements (such as the "1.0 2.0" for the first training vector and the "1.5 2.5" for the first test vector). And, finally, there are three (3) classes of data.

After this first line are all of the training vectors followed by all of the test vectors. For each line of training or test data, the first number is any arbitrary integer—it is ignored by Neural Graphics. Since the first line of data in this example specified two (2) elements

within each training and test vector, the next two (2) numbers are the input data (in floating-point format). The last number is the class to which the vector belongs. The classes must be specified using integers beginning with "1" and continuing sequentially to the number specified in the first line of data—the three (3) in this case.

The classes are specified in the input data for two purposes. For the training vectors, the class is used to adjust the weights of the nodes. The weights are adjusted so that the output node corresponding to the desired class has the greatest numerical value for a given training vector (i.e., three classes will require three output nodes, the node with the greatest numerical value corresponds to the network's estimation of the class: when output-node #1 has the greatest numerical value, the network says the vector belonged to class #1—and so forth for the other two output nodes).

For the test vectors, the class is used to assess the accuracy of the network. At specified intervals, the network will test with the test vectors. For each test vector, the network will compare the network's estimation of the class to the test vector's actual class.

For the example above, lines 1-5 are training vectors and lines 6-11 are test vectors. Neural graphics trains (i.e., adjust the node weights) using the training vectors and tests the accuracy of the neural network using the test vectors. Statistics are produced for the training and testing of the network as shown in the following section.

The data were statistically normalized since neural networks works better when the data are centered about zero and don't deviate more than ± 3.

A.2 Neural Graphics Output Data

For output, Neural Graphics generates the file specified in **setup.fil**. For the example in Section 3.3.4.1, this file would be named **output.data**. This file contains statistics based on the training and testing of the neural network. Neural Graphics generates this file as an average of five separate and complete runs.

As default, Neural Graphics will run 50000 iterations on the training data and will test the network every 2000 iterations. The statistics generated are

- Mean-squared error. A measure of how close the output was numerically to the correct output.
- Training "Rightness." Indicates the percentage of output nodes that were within 20% of their correct values for the training data.
- Training "Goodness." Indicates the percentage of training vectors for which the correct output node (or class) was selected.
- Test Data "Rightness." Indicates the percentage of output nodes that were within 20% of their correct values for the test data.
- Test Data "Goodness." Indicates the percentage of test vectors for which the correct output node (or class) was selected.

The following file is an example of the output statistics generated by Neural Graphics (the column headers were added in this document—they are not generated by Neural Graphics):

Iter	MSE	Trng Rt	Trng Gd	Test Rt	Test Gd
0	4.291	0.000	26.666	0.000	26.666
2000	1.006	88.667	97.333	84.000	95.999
4000	0.617	93.999	98.666	90.667	100.000
6000	0.662	96.667	98.000	93.333	100.000
8000	0.597	96.667	98.666	92.000	95.999
10000	0.501	96.667	98.666	90.667	95.999
12000	0.332	97.333	100.000	90.667	97.333
14000	0.210	100.000	100.000	90.667	100.000
16000	0.186	100.000	100.000	89.333	98.666

18000	0.173	100.000	100.000	90.667	98.666
20000	0.164	100.000	100.000	90.667	98.666
22000	0.154	100.000	100.000	90.667	97.334
24000	0.146	100.000	100.000	92.000	97.334
26000	0.139	100.000	100.000	93.333	97.334
28000	0.129	100.000	100.000	89.333	97.334
30000	0.122	100.000	100.000	89.333	97.334
32000	0.118	100.000	100.000	89.333	97.334
34000	0.113	100.000	100.000	89.333	97.334
36000	0.108	100.000	100.000	89.333	93.333
38000	0.105	100.000	100.000	92.000	93.333
40000	0.102	100.000	100.000	93.333	96.000
42000	0.098	100.000	100.000	92.000	96.000
44000	0.095	100.000	100.000	92.000	96.000
46000	0.092	100.000	100.000	92.000	96.000
48000	0.090	100.000	100.000	92.000	96.000
50000	0.088	100.000	100.000	93.333	96.000

The first column is the iteration count for which that line of statistics was generated. The second column is the mean-square error. The third column is the training "rightness." The fourth column is the training "goodness." The fifth column is the test data "rightness." The sixth column is the test data "goodness."

As shown above, at iteration 6000, the training data was accurately sorted into the correct class 98% of the time; whereas, the test data was accurately sorted into the correct class 100% of the time. Iteration 14000 was ideal: both the training and test data were accurately sorted into the correct class 100% of the time. At this point, the node weights would be saved and used for future processing.

A.3 Specifying Training and Test Vectors

For this research, synthetic imagery was generated for every one-degree of sensor view-angle (360 images for each sensor depression-angle). The features were extracted from these images and the resulting numbers formatted for processing by Neural Graphics. In this section, the conventions used in formatting the data for this processing are presented.

In the input files, for data from a single sensor depression-angle viewpoint, the training and test vectors were selected so that there would be twice as many training vectors as there were test vectors. To this end, the vectors were sorted so that the first two vectors were training vectors, the next single vector was a test vector, the following two vectors were training vectors, the next single vector was a test vector, and so forth:

```
sensor
view-      training/
angle      test
-------    ---------
  0        training
  1        training
  2        test
  3        training
  4        training
  5        test
  etc.
```

For comparing the robustness of the network trained with data at one sensor depression-angle for use in discriminating data from a different sensor depression-angle, all data from one depression angle was used to train the net while all data from the other depression angle was used to test the net. Results are presented in Chapter 4.

A.4 Specifying Orientation Angles

For the input files, the first column of vector data (the column which is not processed by Neural Graphics) was used to identify the sensor view-angle:

```
8 4 2 3
0  0.0 2.9  1  <-- start of training vectors
1  1.0 2.0  1
3  1.1 2.1  1
4  1.3 2.3  2
6  1.4 2.4  2
7  1.6 2.6  2
9  1.7 2.7  3
10 1.9 2.9  3
2  1.9 2.9  1  <-- start of test vectors
5  1.2 2.2  2
8  1.8 2.8  3
11 1.9 2.9  3
```

In the above example, the sensor view-angles were 0 degrees through to 11 degrees (0, 1, 3, 4, 6, 7, 9, 10, 2, 5, 8, 11).

A.5 Specifying Angle-Bin

The input files were created assuming the target was placed into the correct angle-bin from the previous hierarchical angle-bin sorting. The results from Neural Graphics' processing of the data would give the accuracy for just the sorting indicated in the input file. For example, for determining the 15-degree angle-bin for a target given the correct 90-degree angle-bin, it was assumed the 90-degree angle-bin was chosen with 100% accuracy; the output of Neural Graphics would give the accuracy of the 15-degree sorting within that specific 90-degree angle-bin.

The classes specified in the input file corresponded to the angle-bin sought. For example, in sorting the orientation angle into 5-degree angle-bins given the correct 15-degree angle-bin, the data would have been formatted as follows (there are three 5-degree angle-bins in a given 15-degree angle-bin; therefore, for 0-14 degrees as the 15-degree angle-bin, there would be three classes—one for 0-4 degrees, one for 5-9 degrees, and one for 10-14 degrees):

```
10 5 4 3
0   x x x x  1    <-- training data
1   x x x x  1
3   x x x x  1
4   x x x x  1
6   x x x x  2
7   x x x x  2
9   x x x x  2
10  x x x x  3
12  x x x x  3
13  x x x x  3
2   x x x x  1    <-- test data
5   x x x x  2
8   x x x x  2
11  x x x x  3
14  x x x x  3
```

where the x's represent the floating-point numerical features extracted from the image at the sensor view-angle specified (NOTE: the arrows and text indicating the beginning of training and test data were not included in the actual input files—they are shown here only to help with this explanation).

This method of using the classes to indicate the angle-bin was used for all angle-bin sorting. For sorting into 180-degree angle-bins, two classes were used (one for the 0-179

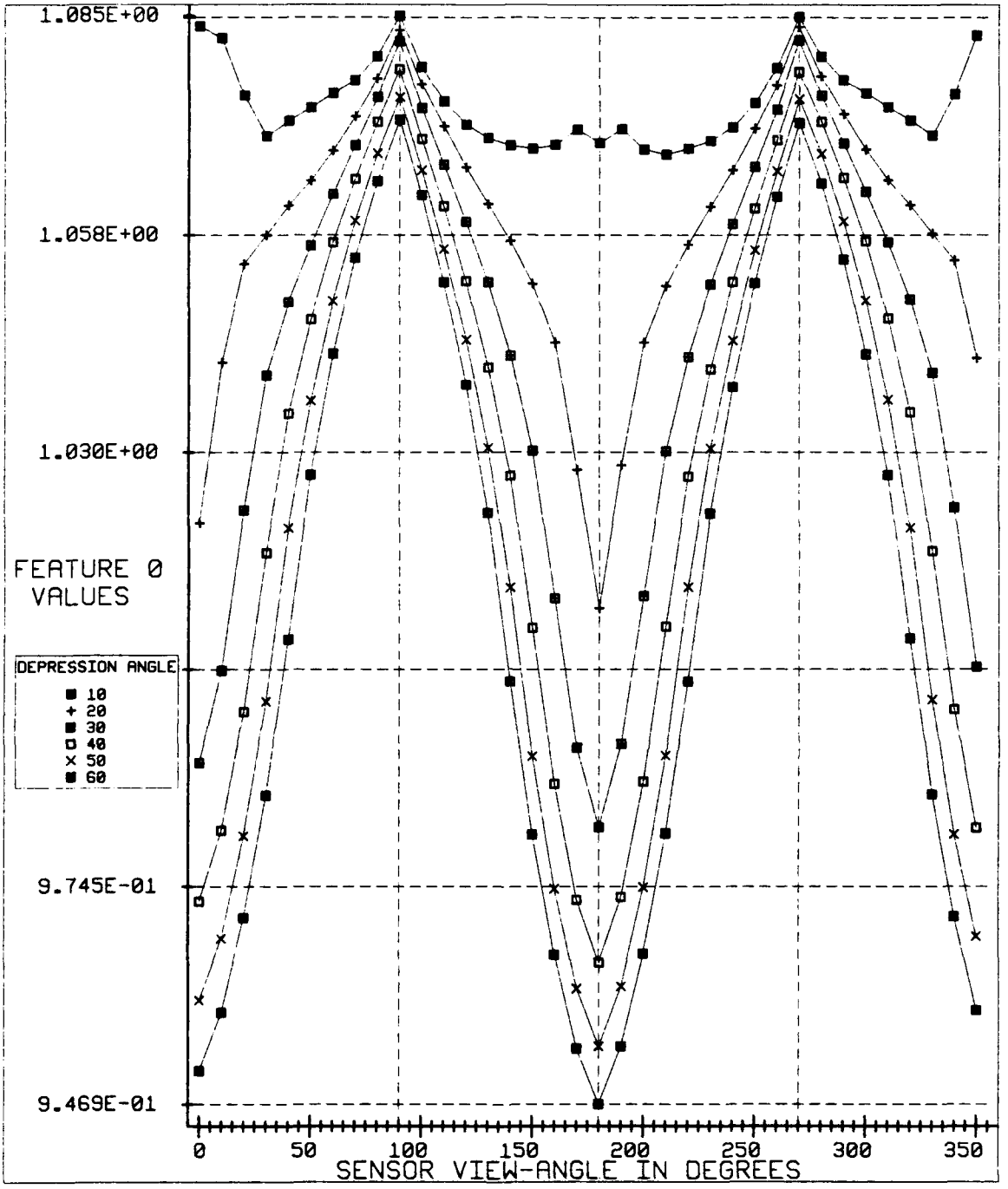
degree angle-bin and one for the 180-359 degree angle-bin). For sorting into 15-degree angle-bins given the correct 90-degree angle-bin, six classes were used, etc.

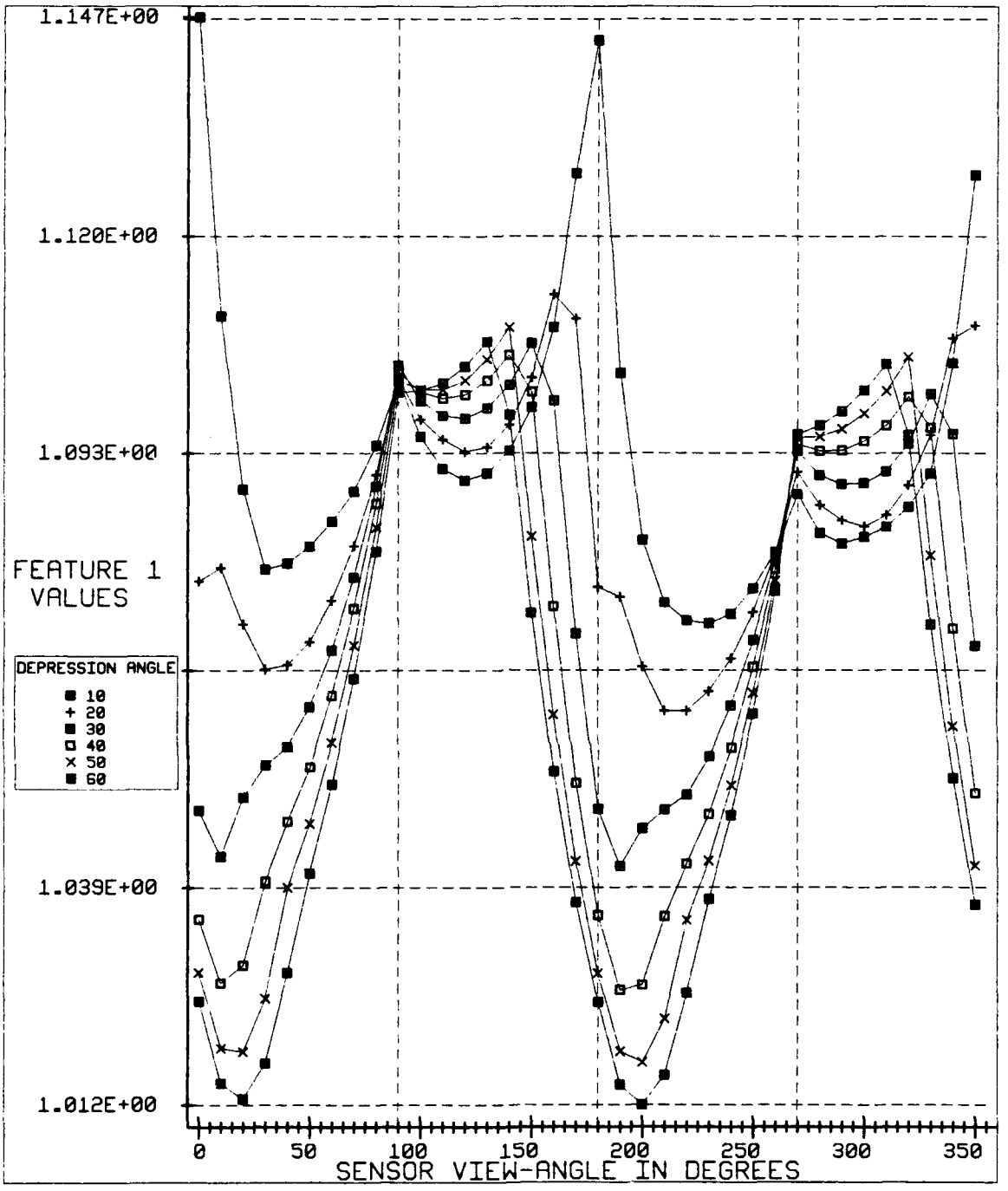
Appendix B. *Additional Data Plots*

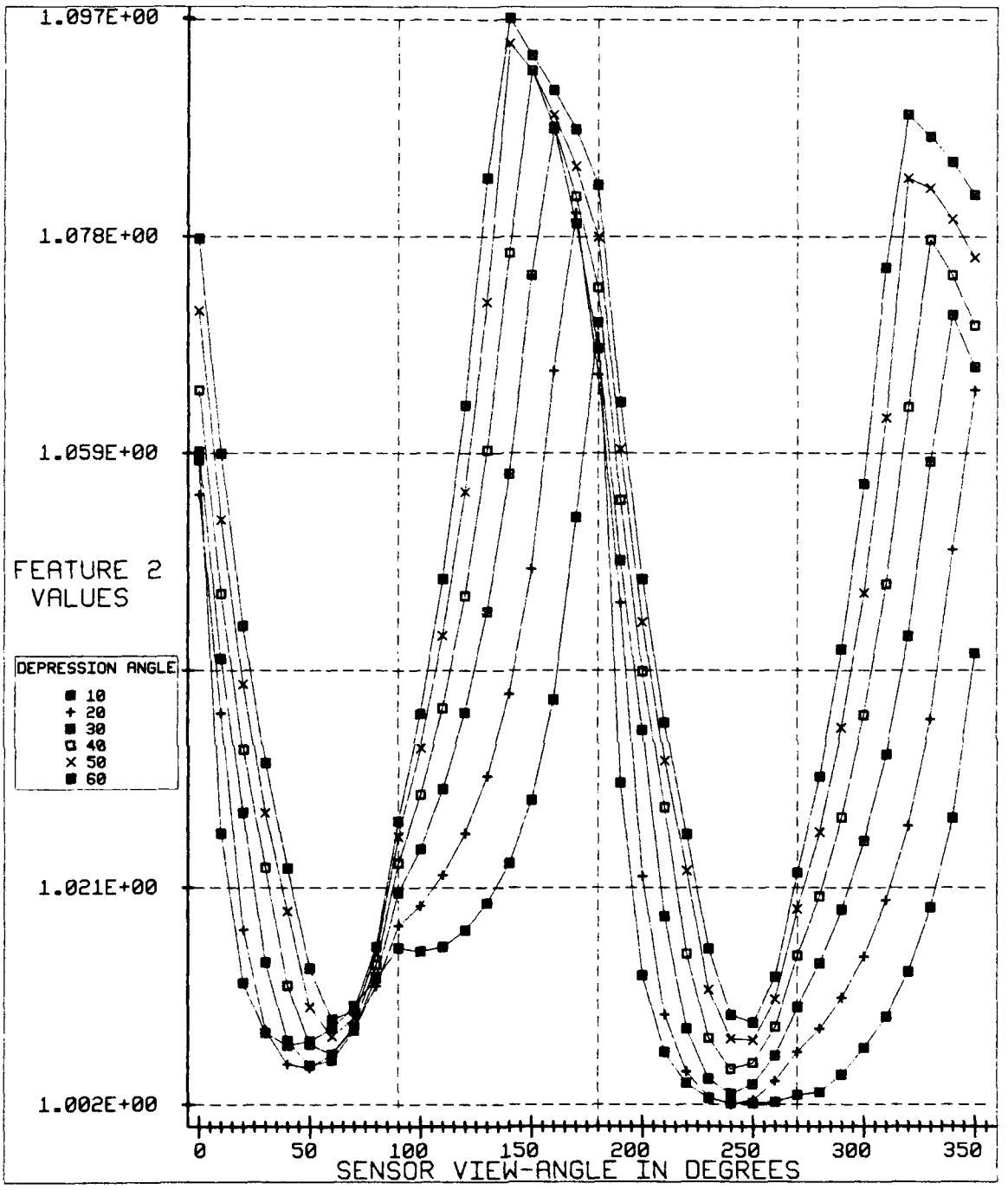
In the following two sections, the feature plots and scale invariance test plots for the entire 36 features are provided.

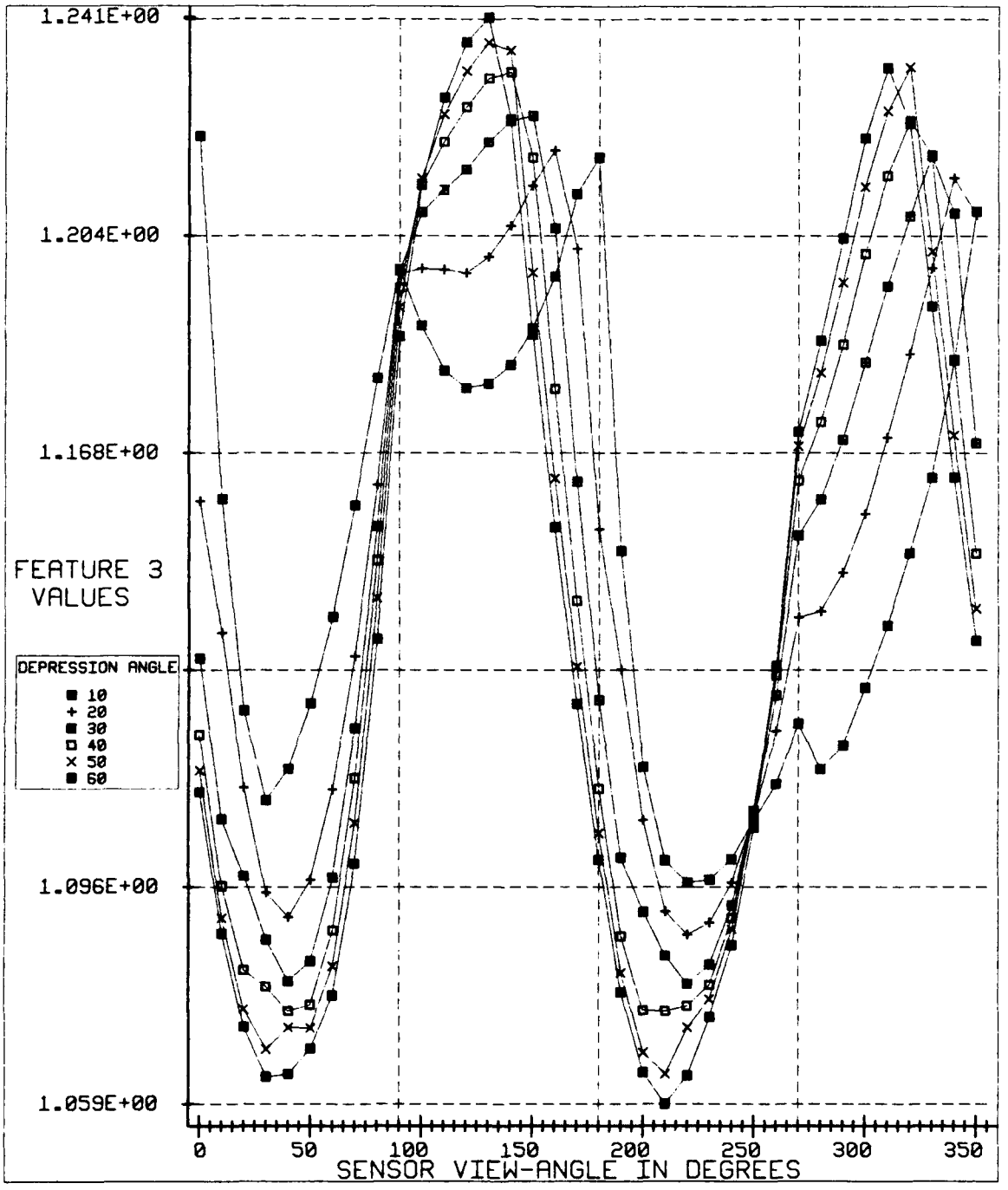
B.1 Feature Plots

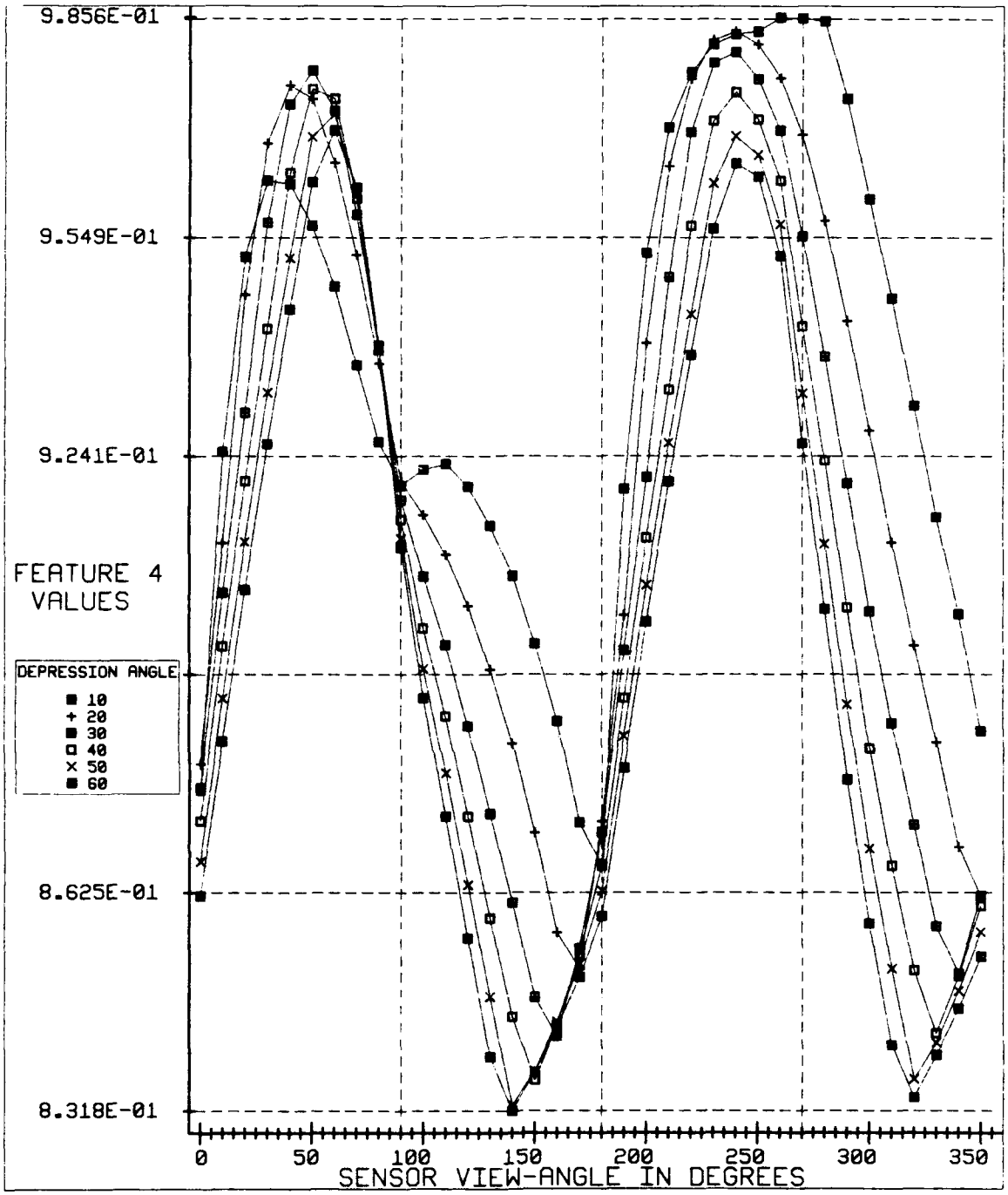
The following 36 graphs are of the 36 features as numbered in Chapter 4. These plots show how the feature values change as the synthetic T-62 is viewed from sensor depression-angle of 10, 20, 30, 40, 50 and 60 degrees. Additionally, for each sensor depression angle, the sensor view-angles from 0 to 350 degrees (inclusive) are plotted at 10 degree increments.

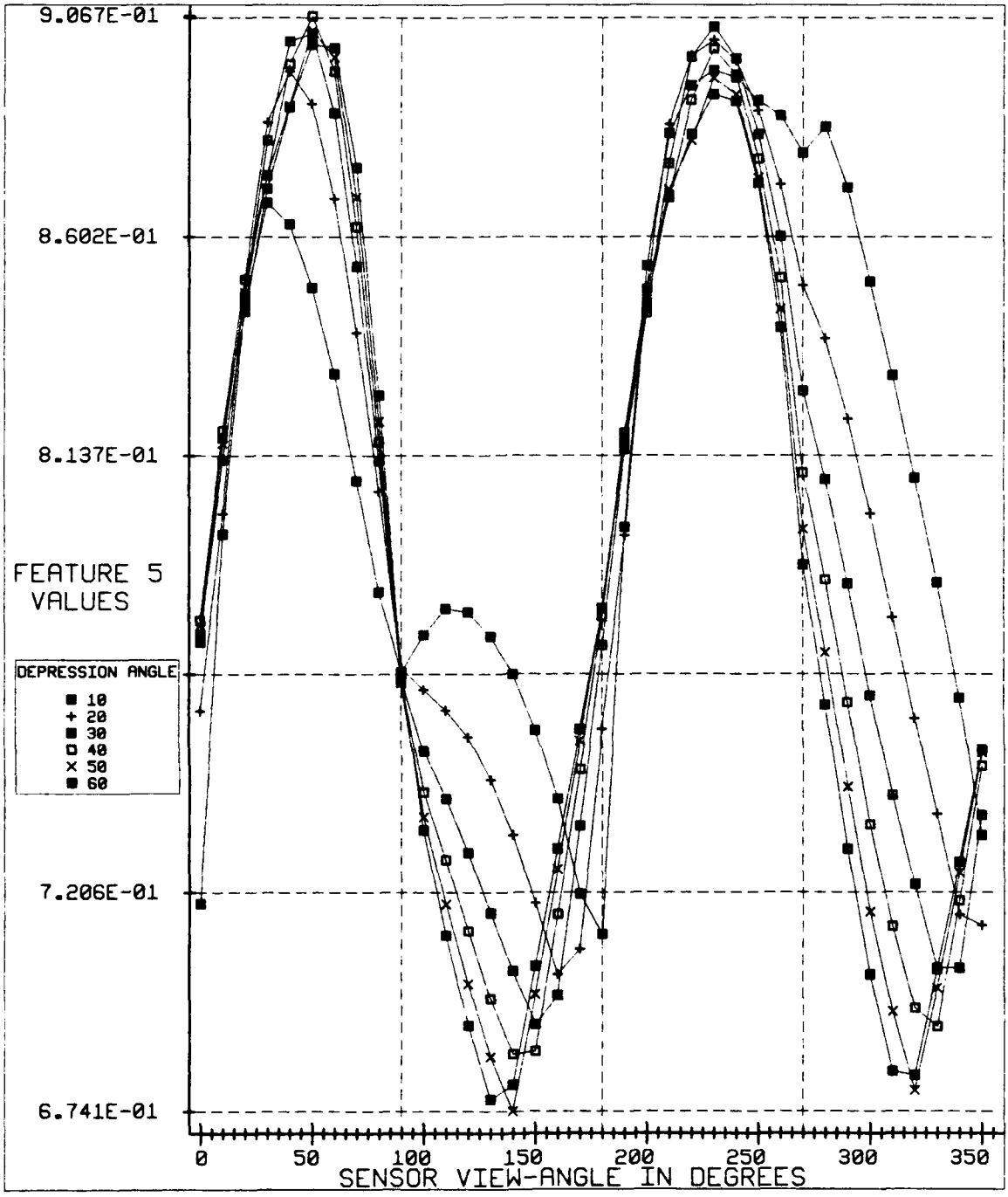

1.085E+00
1.058E+00
1.030E+00
FEATURE 0
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
9.745E-01
9.469E-01
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

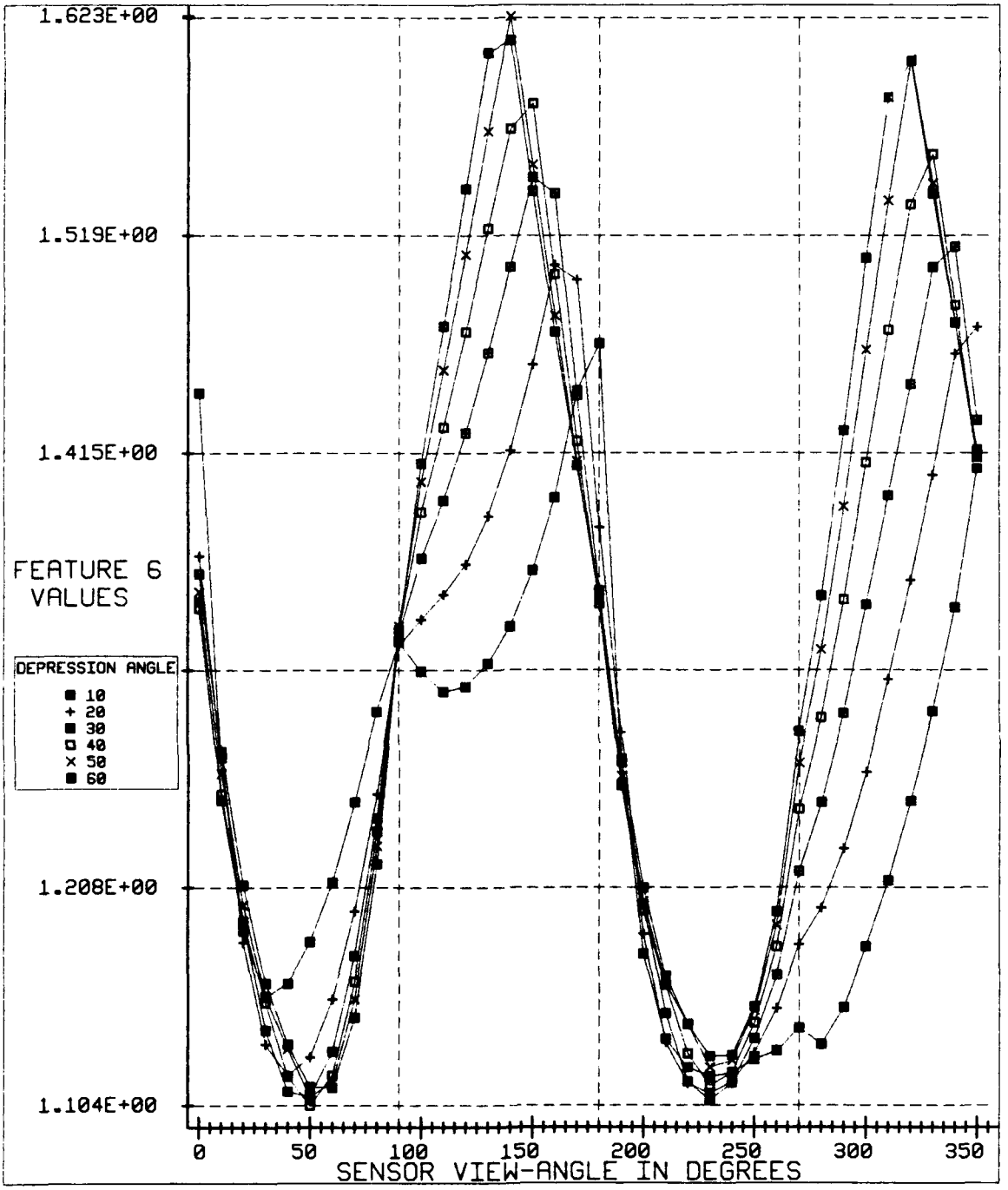

1.147E+00
1.120E+00
1.093E+00
1.039E+00
1.012E+00
FEATURE 1
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

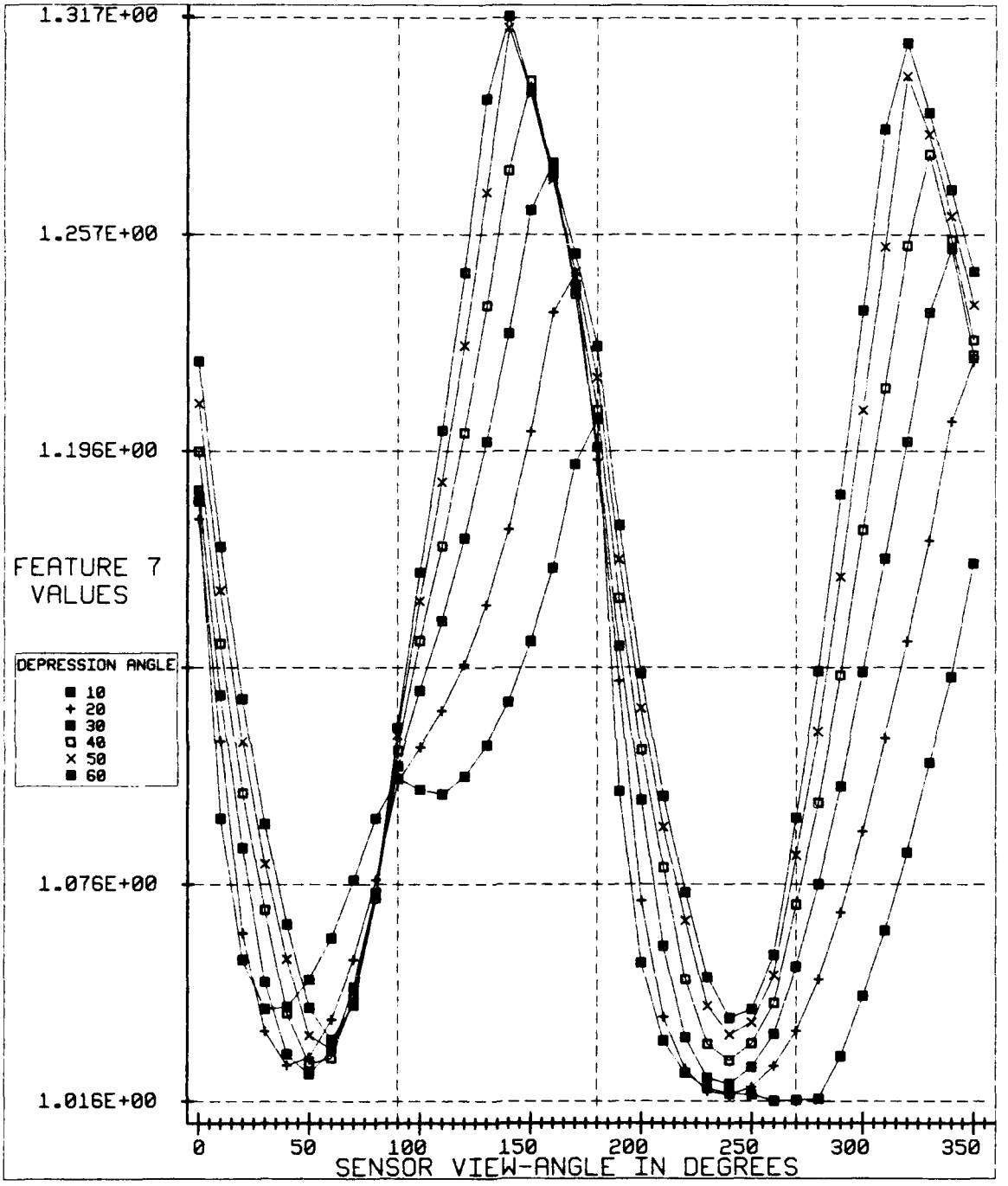

1.097E+00
1.078E+00
1.059E+00
1.021E+00
1.002E+00
FEATURE 2
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

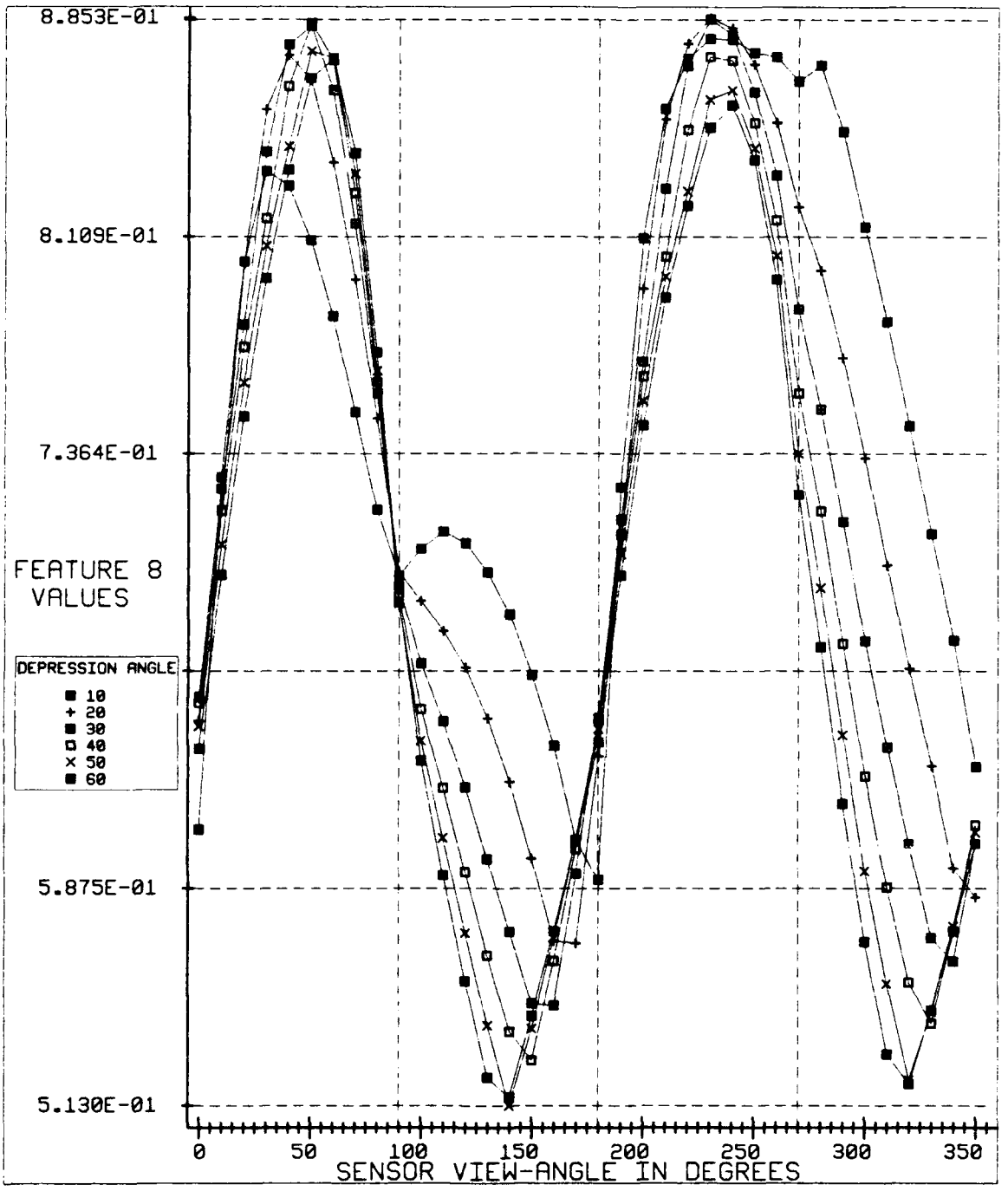

1.241E+00
1.204E+00
1.168E+00
1.096E+00
1.059E+00
FEATURE 3
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

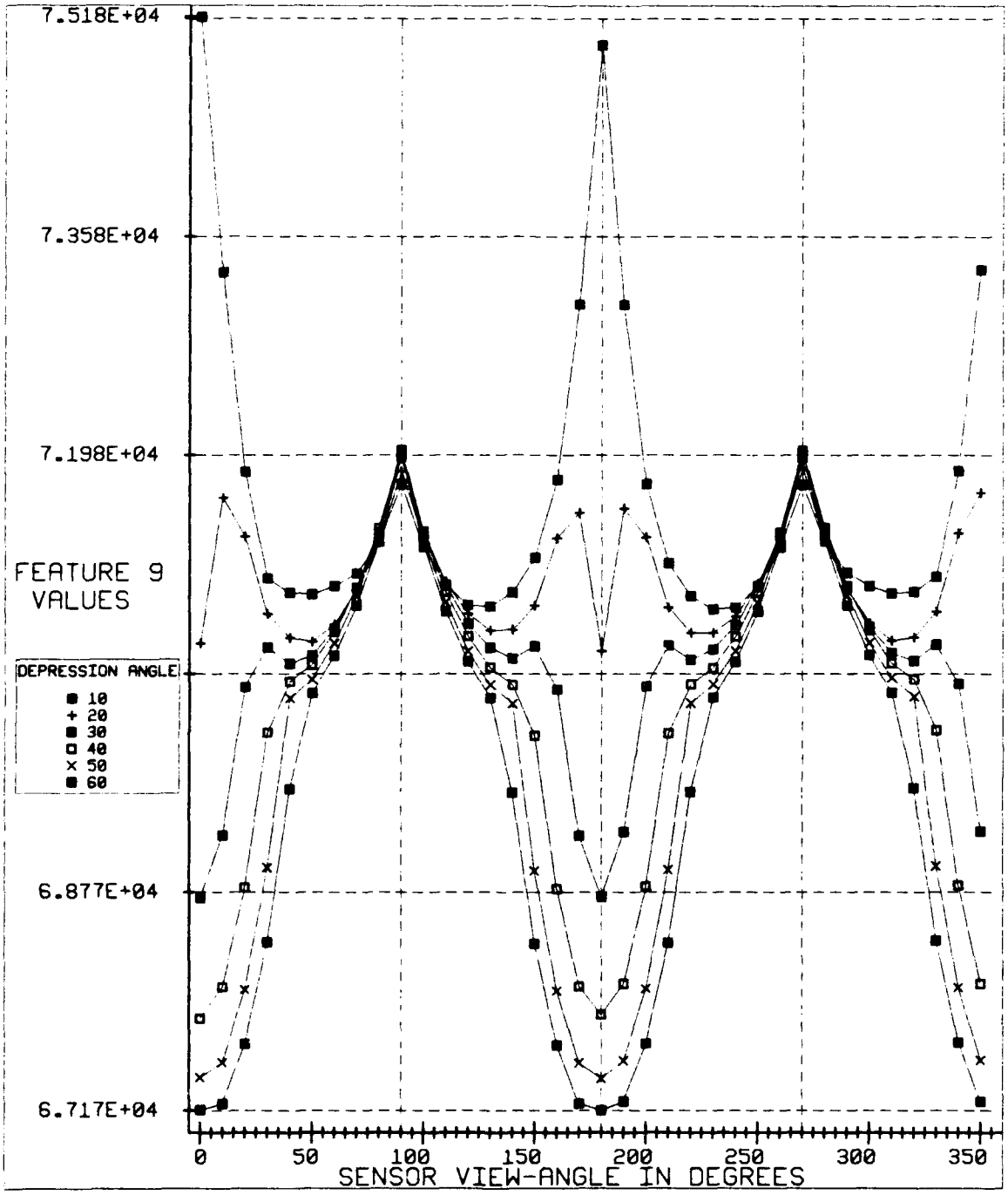

9.856E-01
9.549E-01
9.241E-01
FEATURE 4
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
8.625E-01
8.318E-01
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

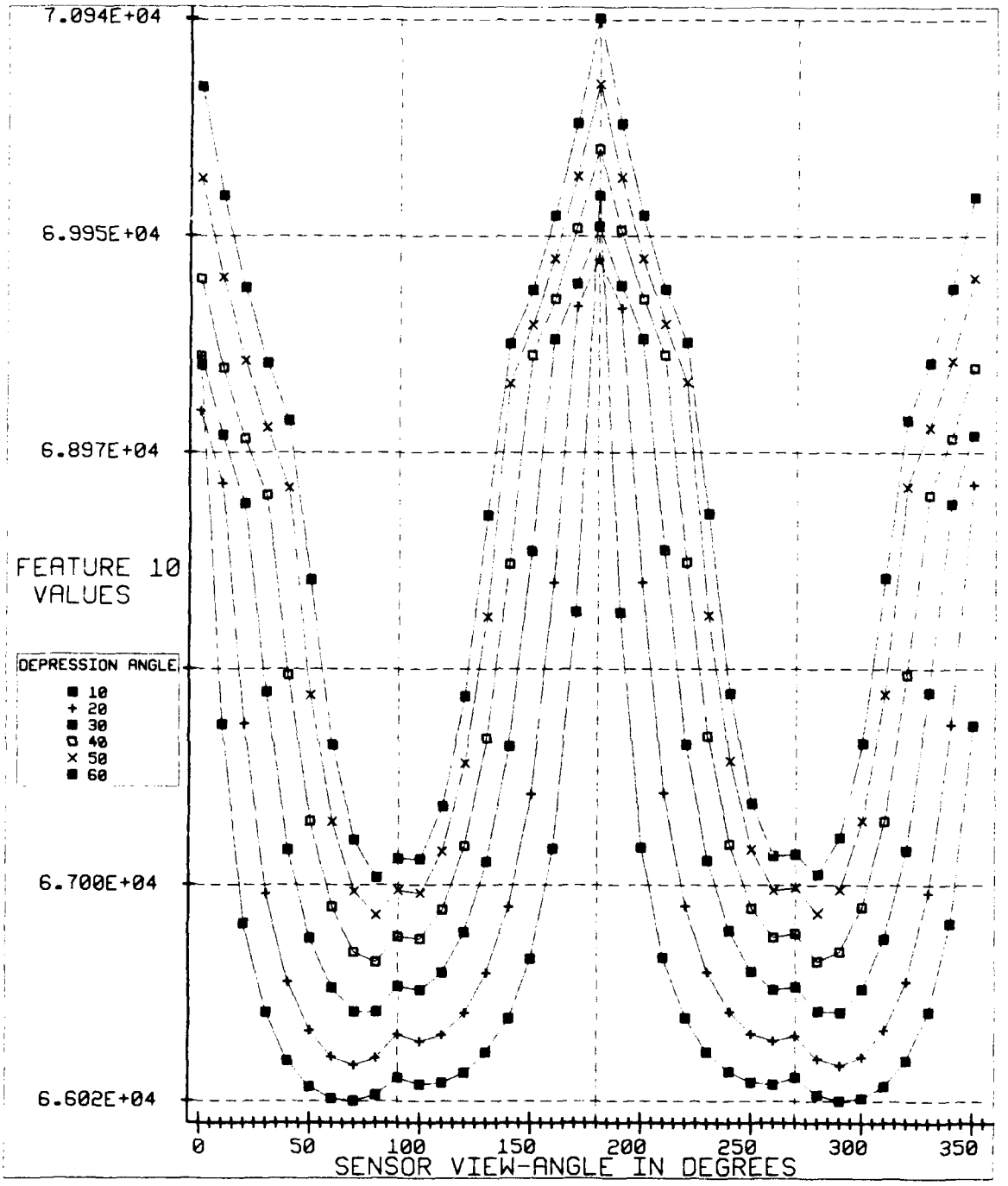

9.067E-01
8.602E-01
8.137E-01
7.206E-01
6.741E-01
FEATURE 5
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

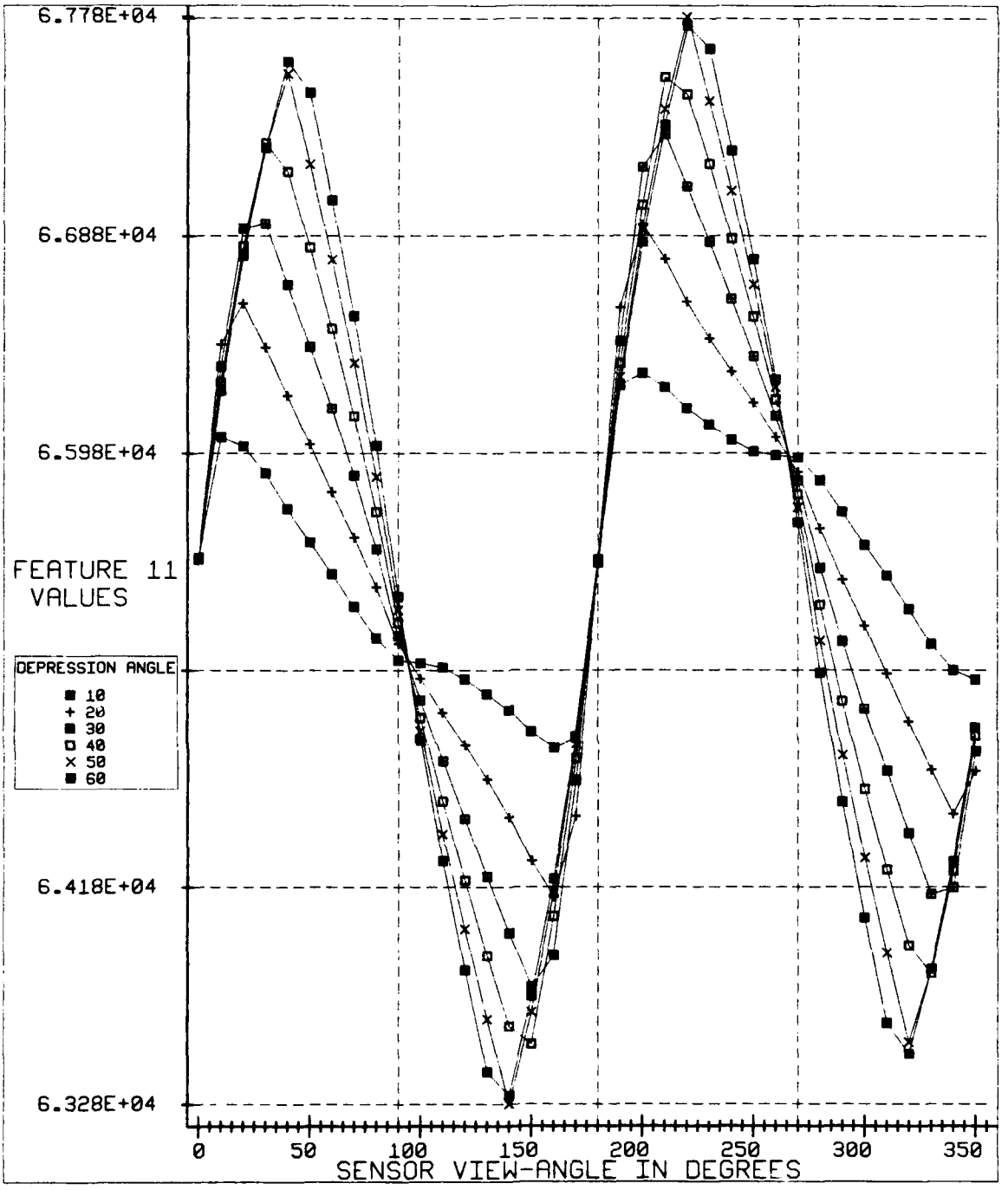

1.623E+00
1.519E+00
1.415E+00
1.208E+00
1.104E+00
FEATURE 6
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

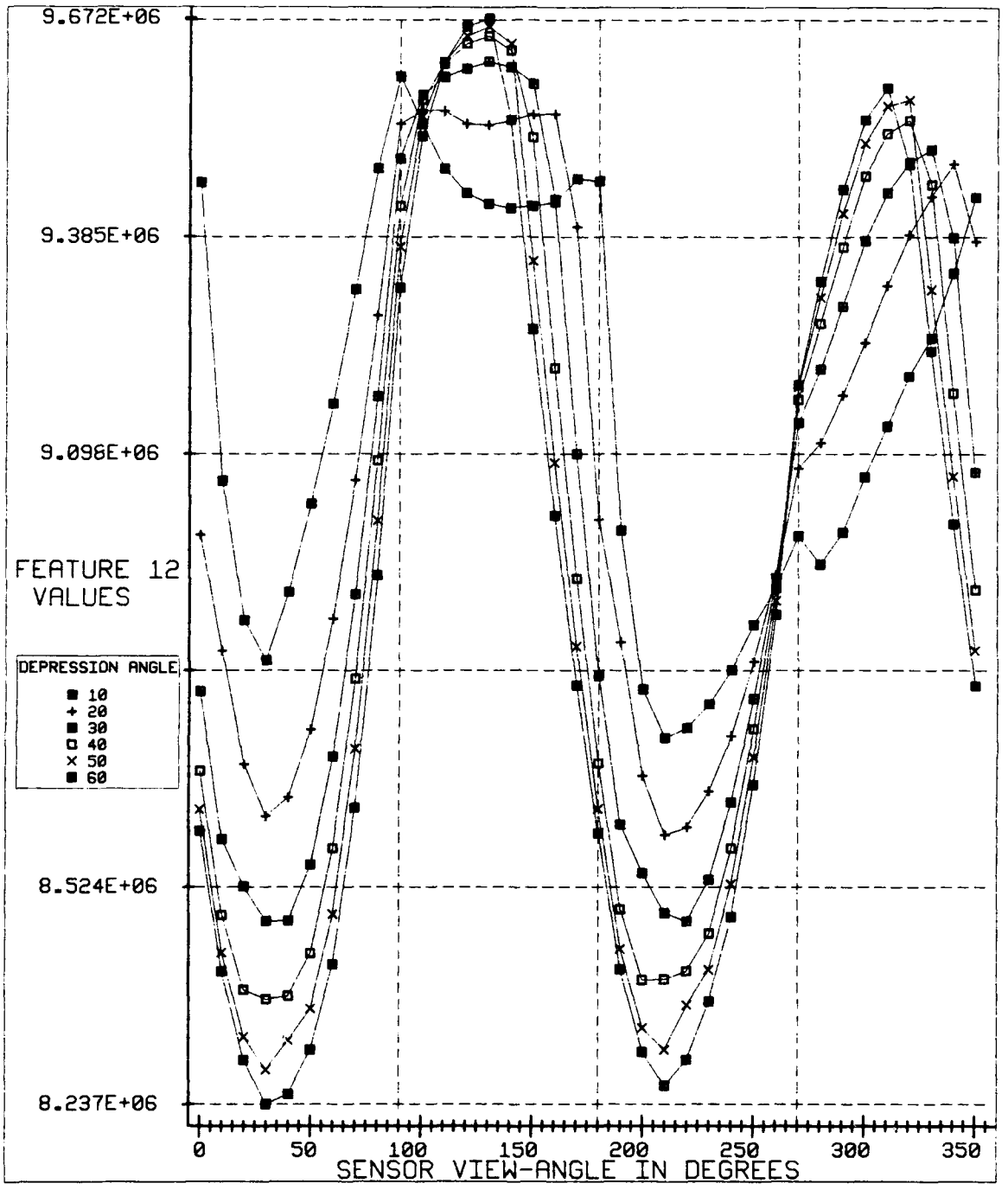

1.317E+00
1.257E+00
1.196E+00
1.076E+00
1.016E+00
FEATURE 7
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

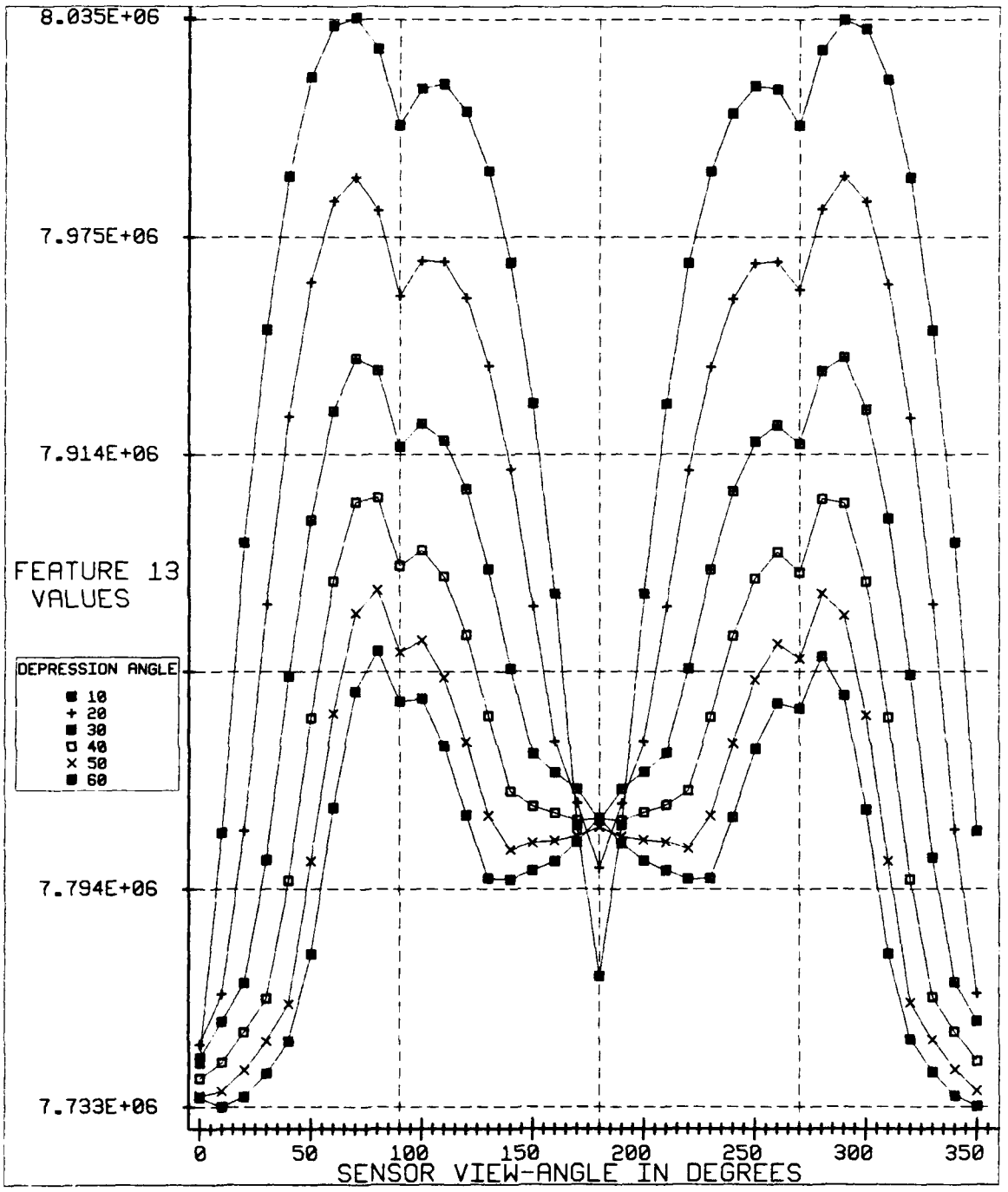

8.853E-01
8.109E-01
7.364E-01
5.875E-01
5.130E-01
FEATURE 8
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

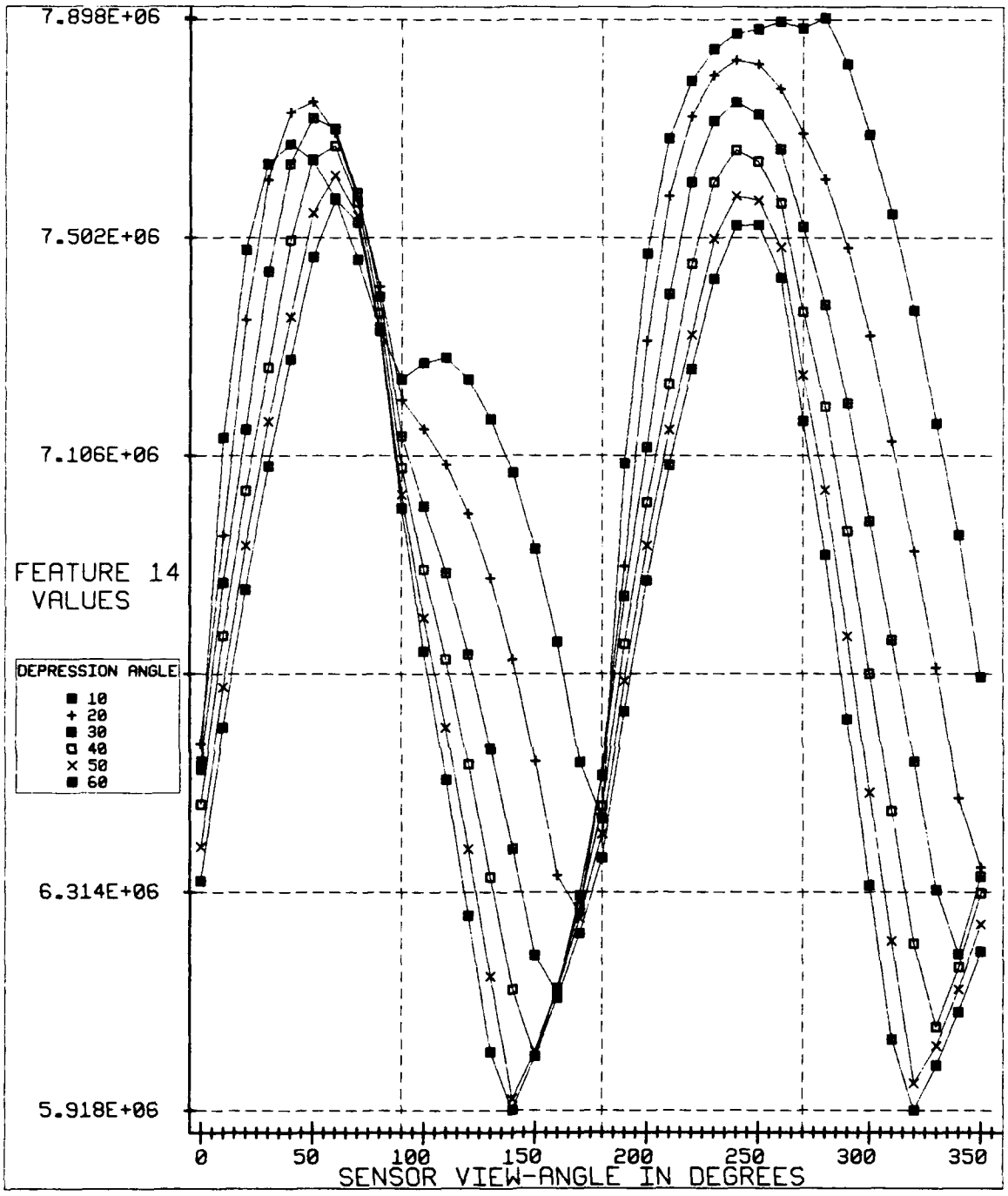

7.518E+04
7.358E+04
7.198E+04
FEATURE 9
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
6.877E+04
6.717E+04
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

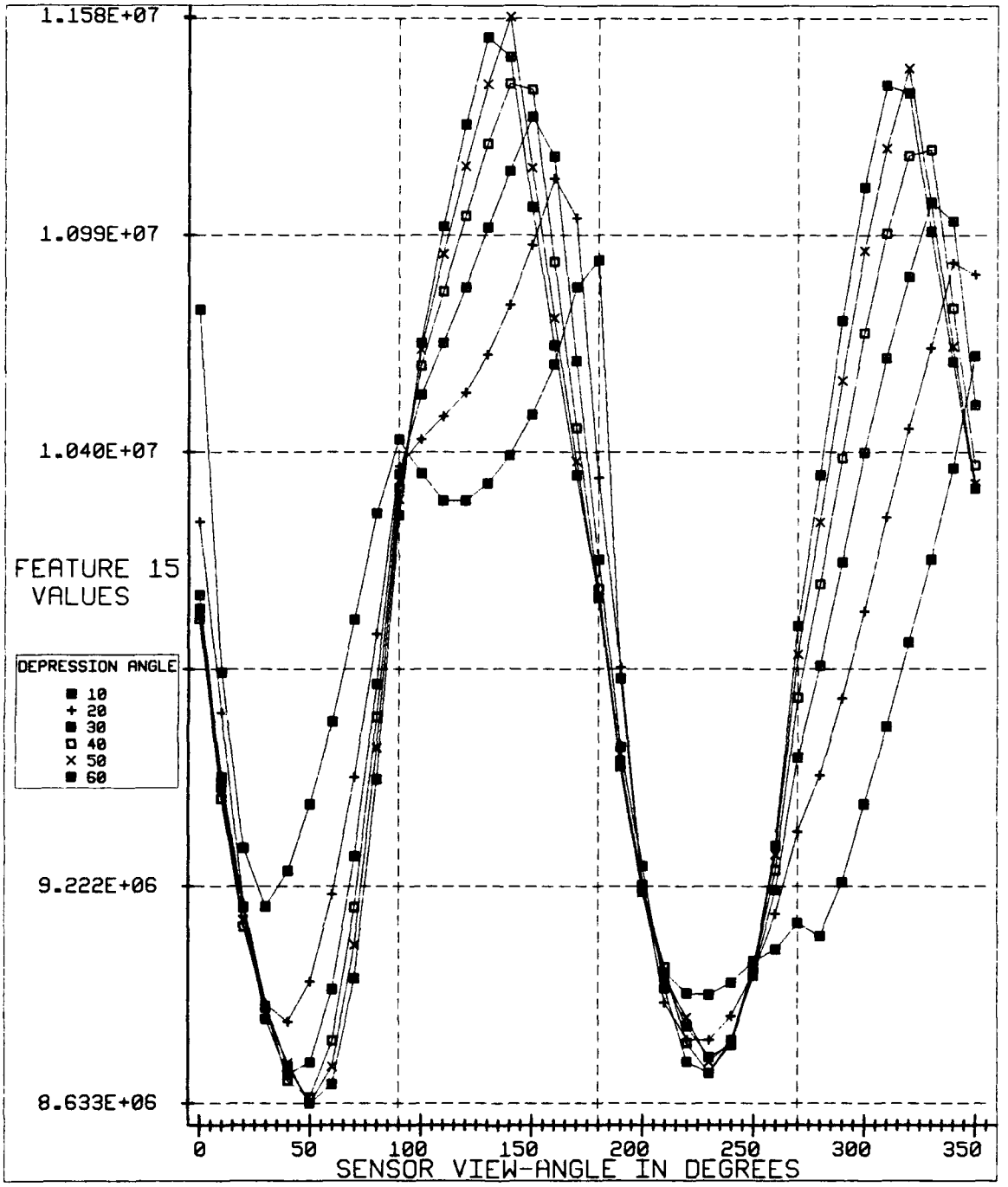

7.094E+04
6.995E+04
6.897E+04
6.700E+04
6.602E+04
FEATURE 10
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

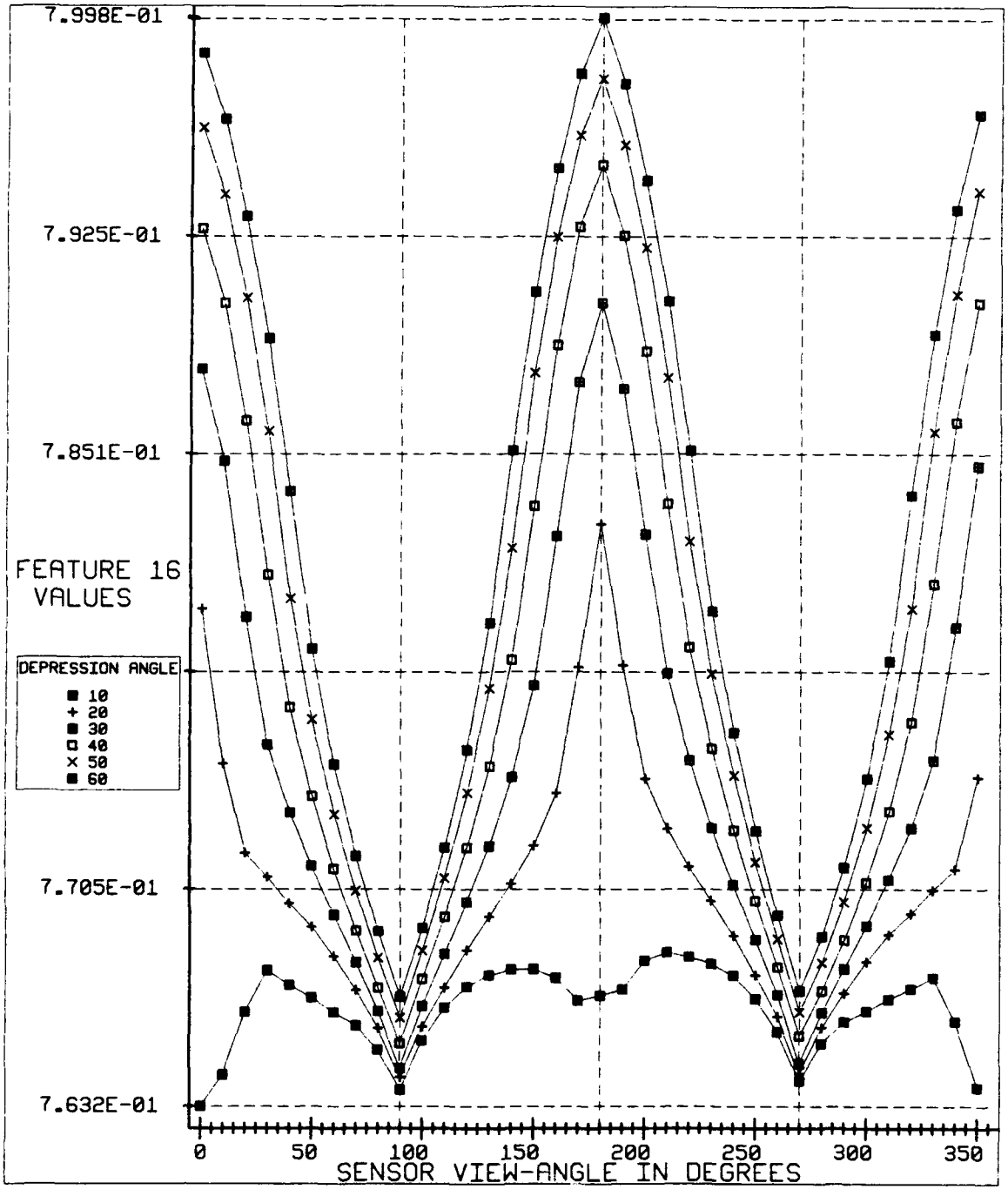

6.778E+04
6.688E+04
6.598E+04
6.418E+04
6.328E+04
FEATURE 11
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

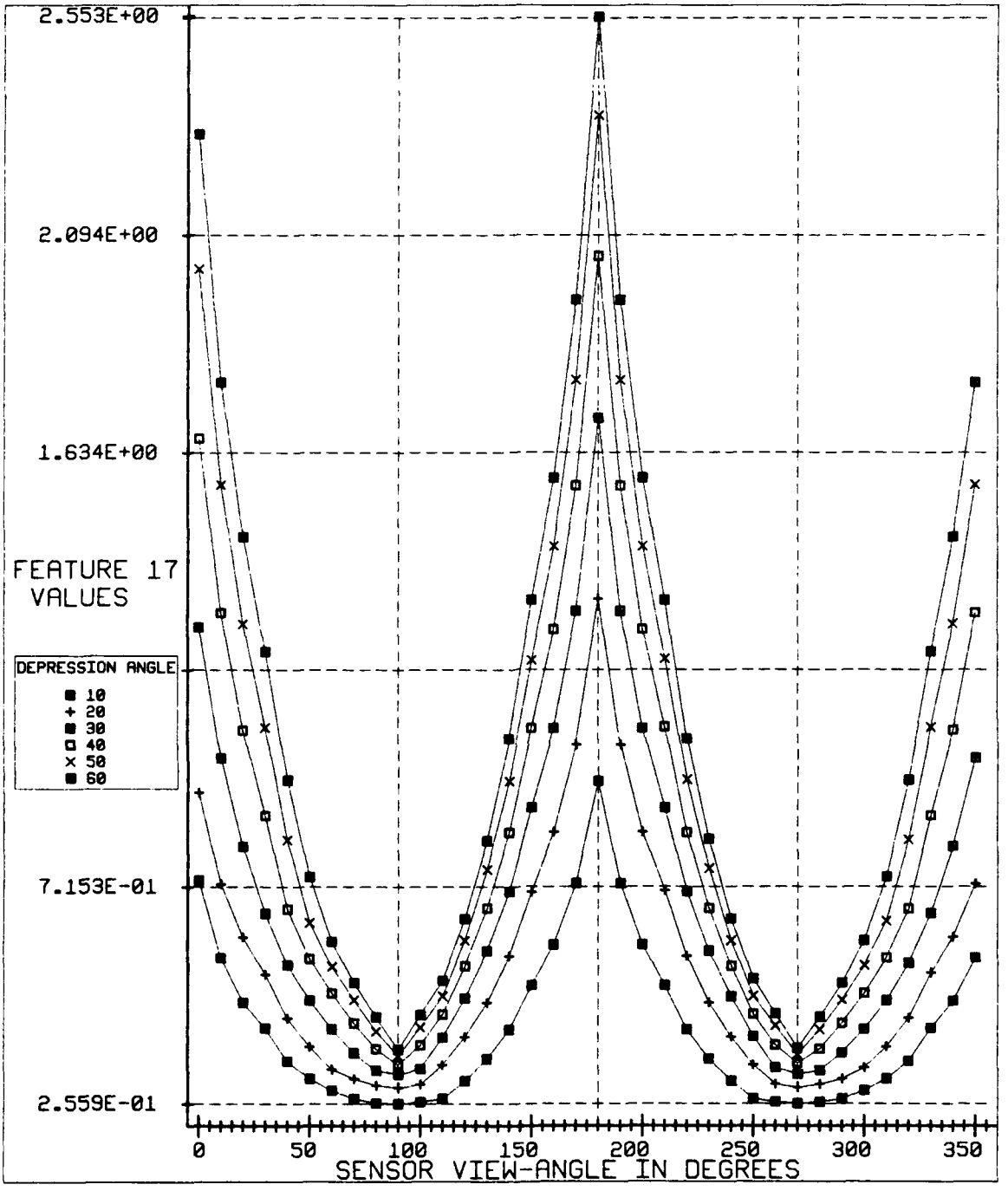

9.672E+06
9.385E+06
9.098E+06
8.524E+06
8.237E+06
FEATURE 12
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

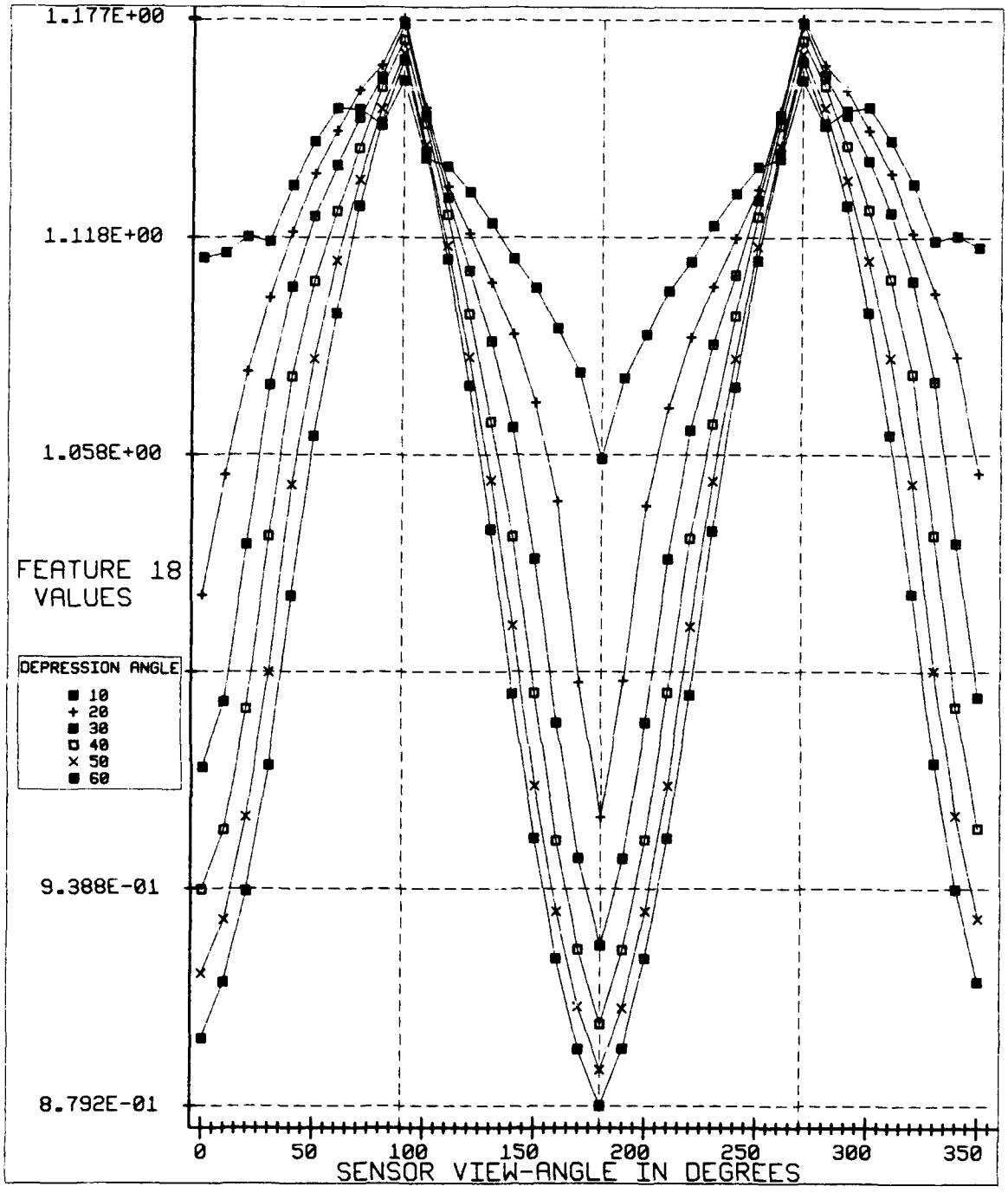

8.035E+06
7.975E+06
7.914E+06
7.794E+06
7.733E+06
FEATURE 13
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

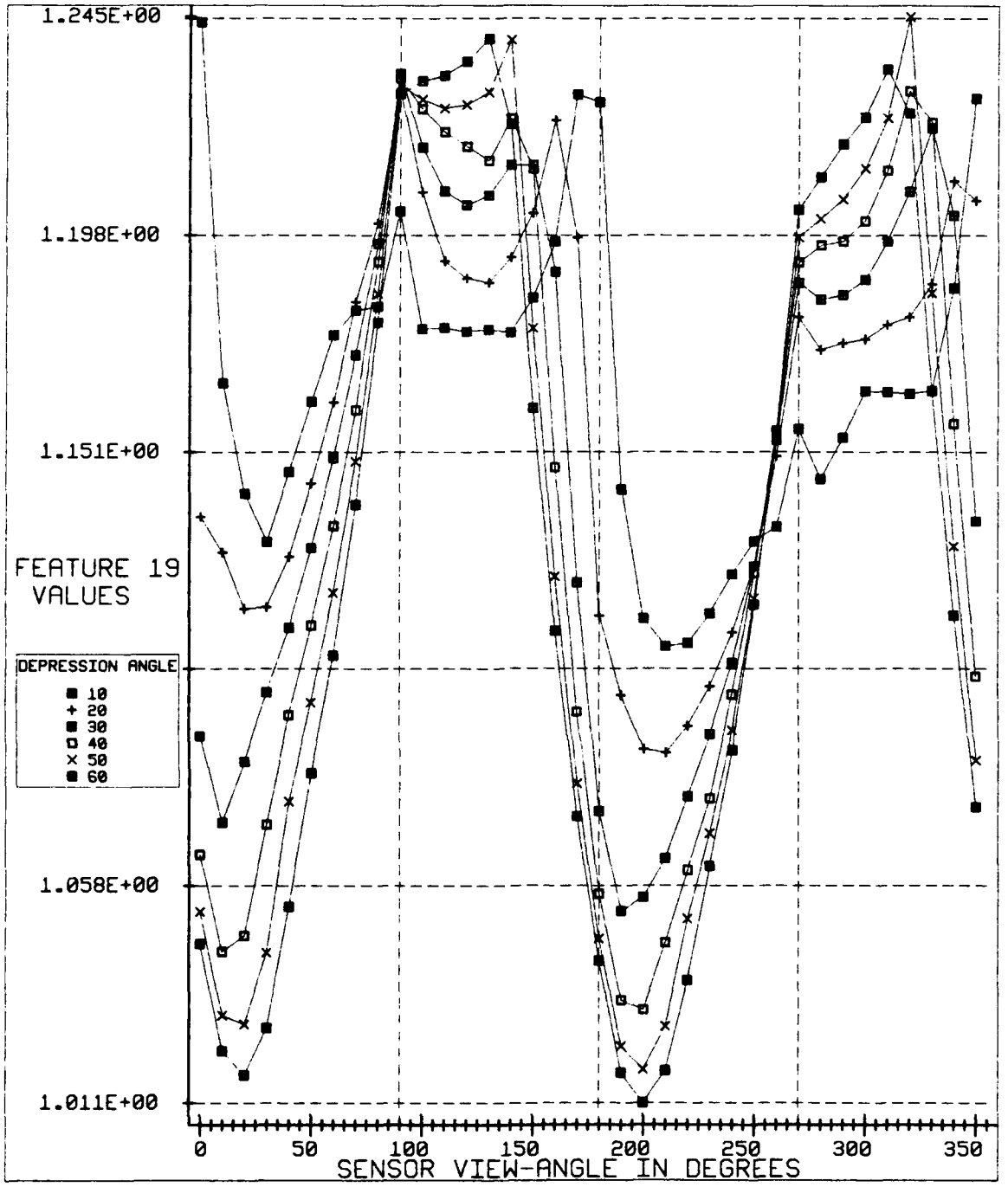

7.898E+06
7.502E+06
7.106E+06
6.314E+06
5.918E+06
FEATURE 14
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

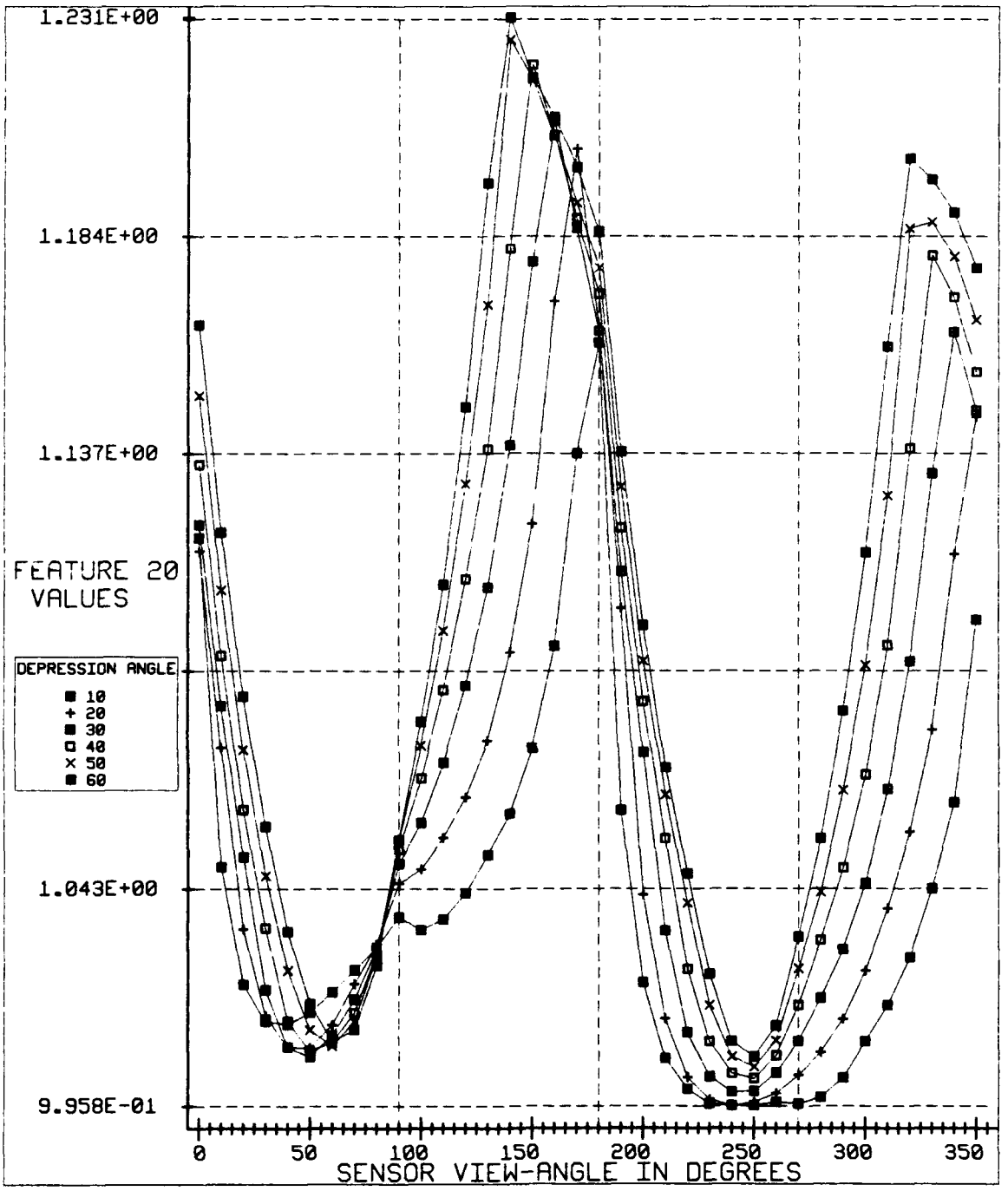

1.158E+07
1.099E+07
1.040E+07
9.222E+06
8.633E+06
FEATURE 15
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

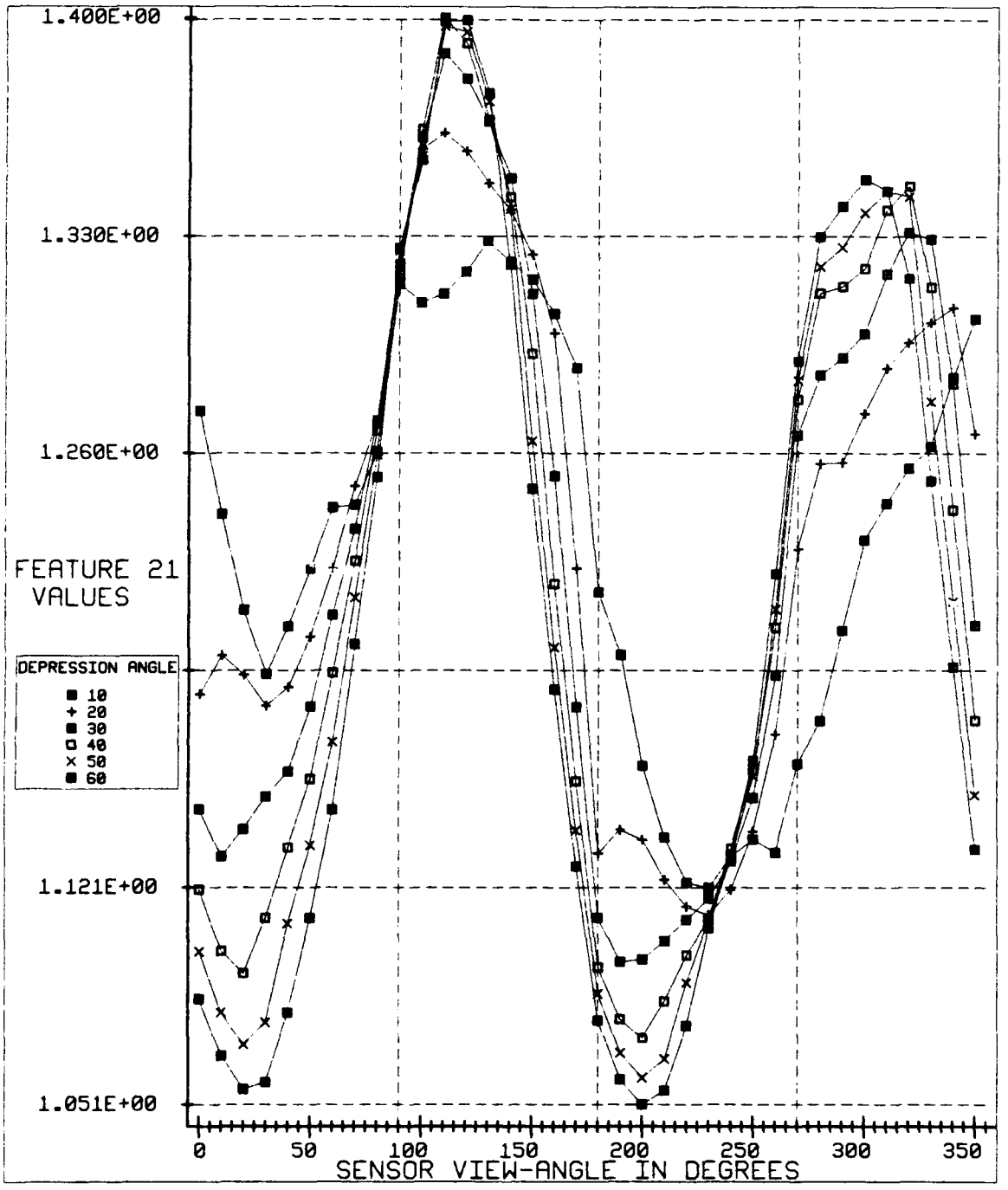

7.998E-01
7.925E-01
7.851E-01
7.705E-01
7.632E-01
FEATURE 16
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

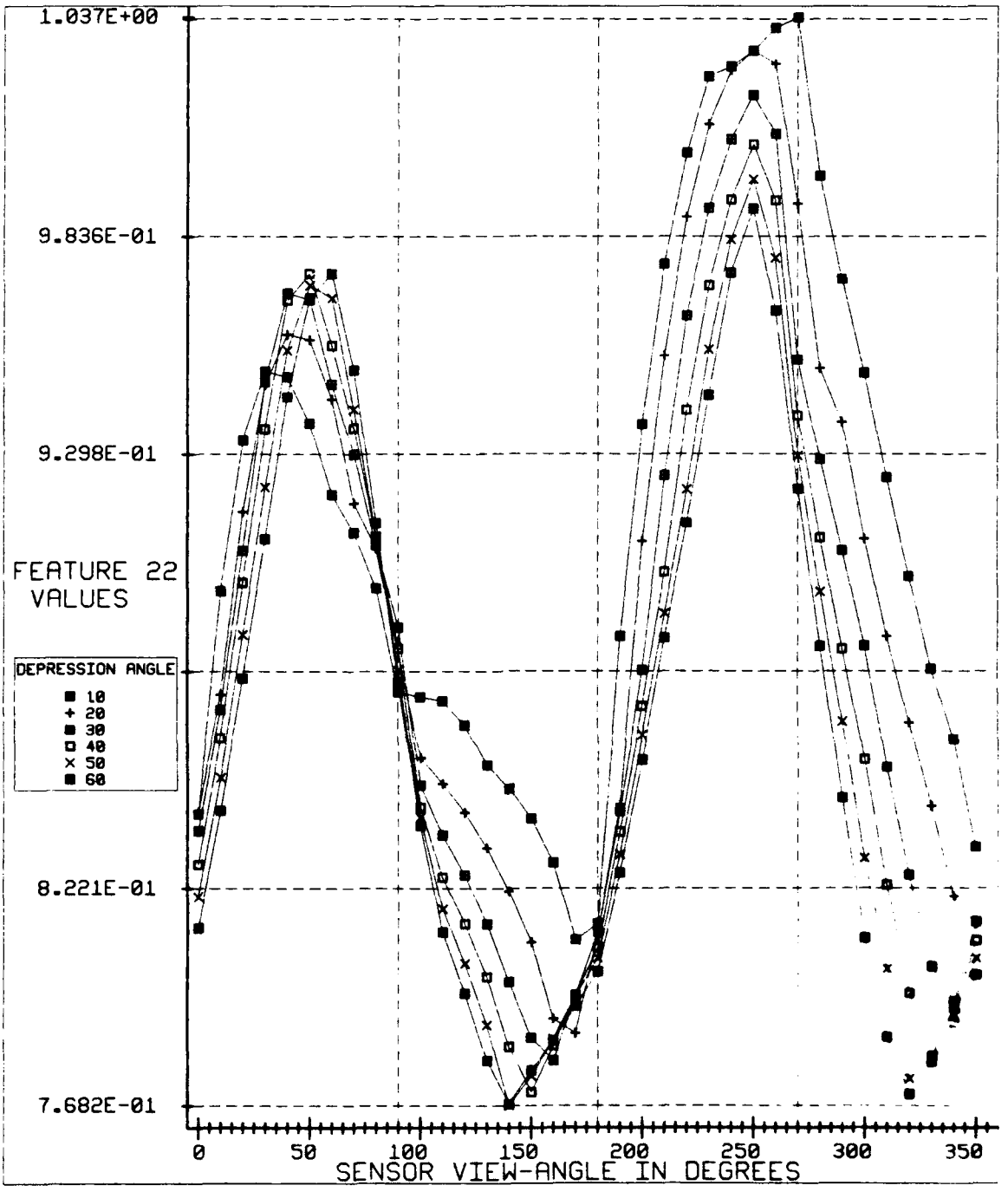

2.553E+00
2.094E+00
1.634E+00
7.153E-01
2.559E-01
FEATURE 17
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

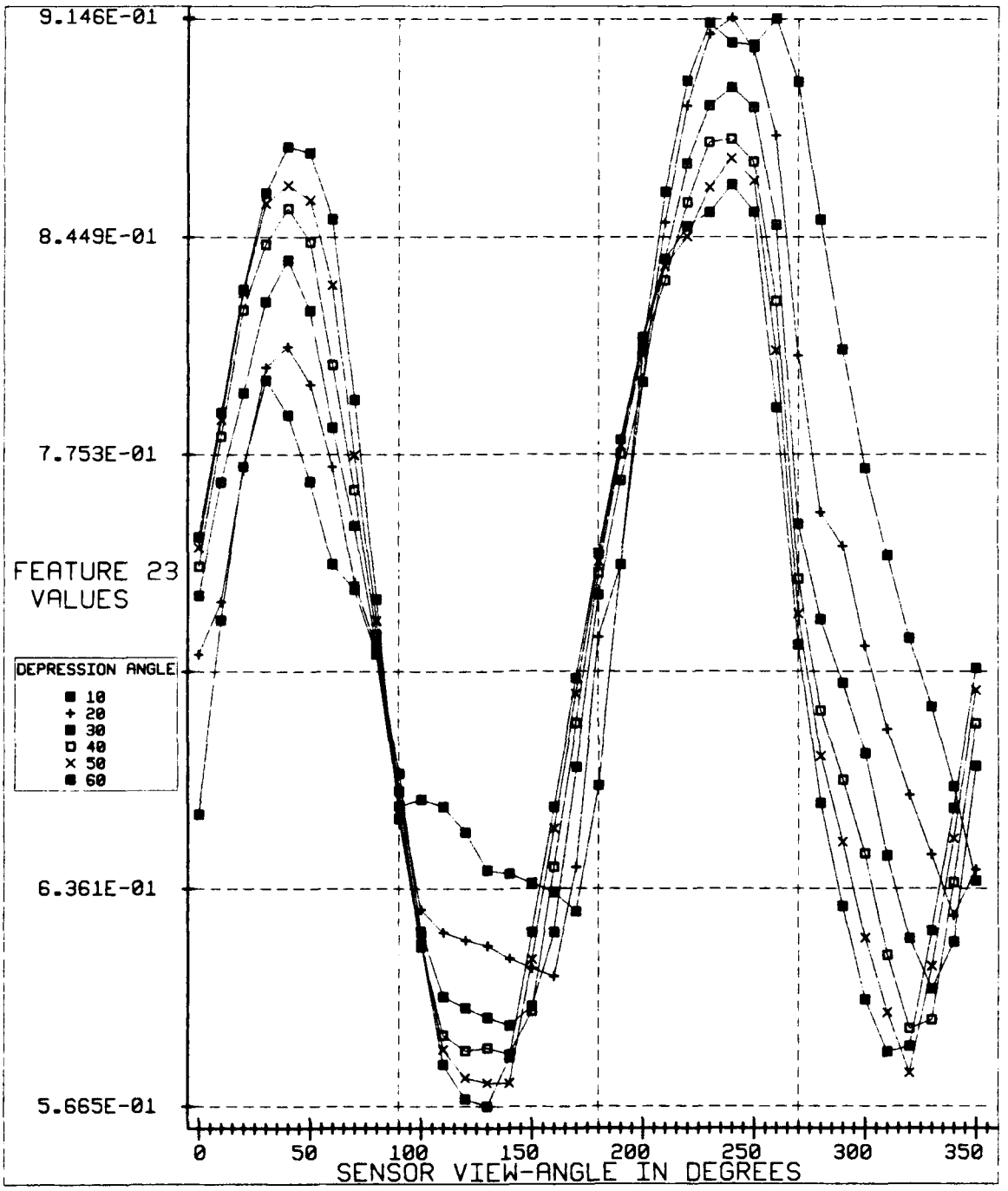

1.177E+00
1.118E+00
1.058E+00
FEATURE 18
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
9.388E-01
8.792E-01
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

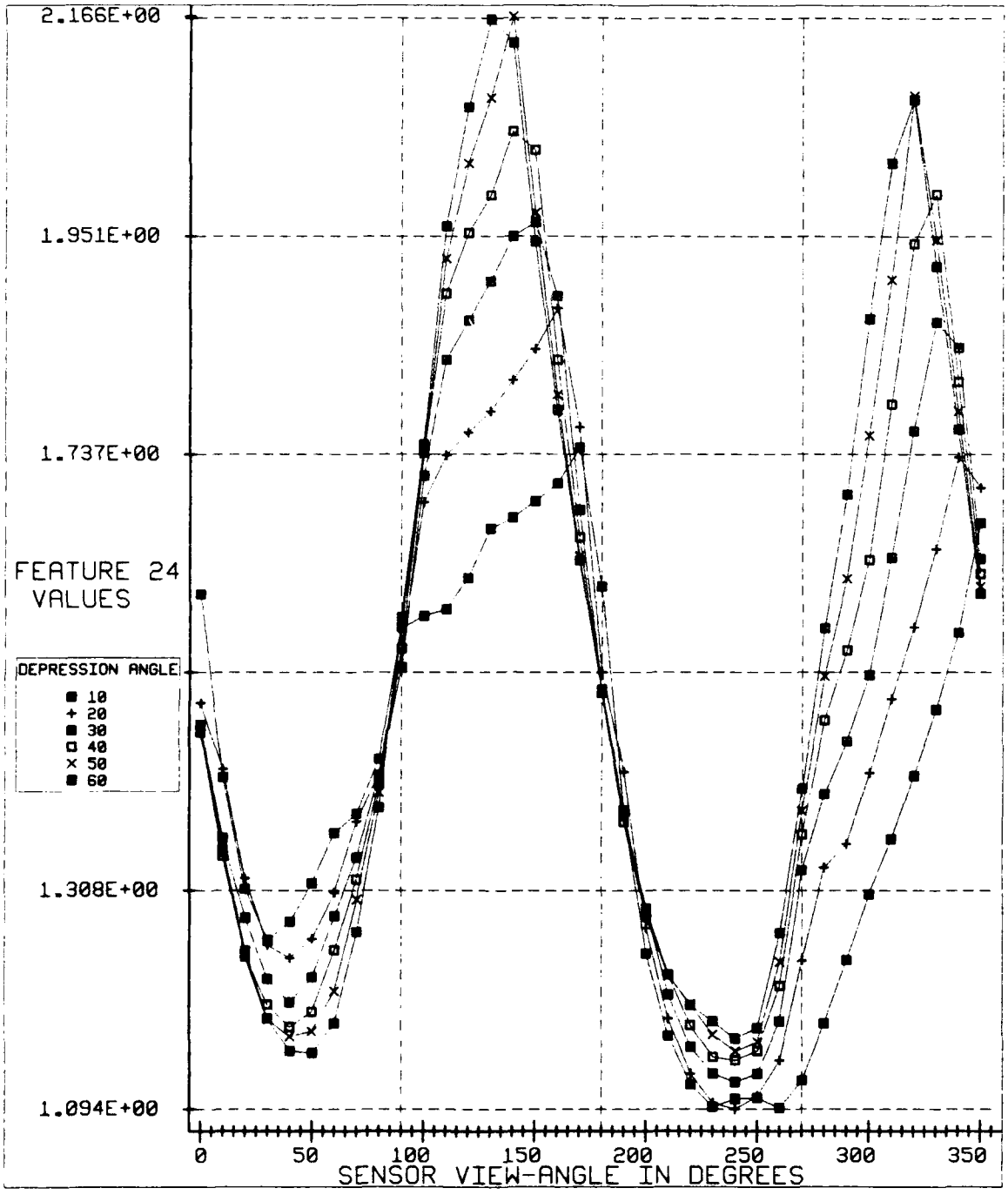

1.245E+00
1.198E+00
1.151E+00
1.058E+00
1.011E+00
FEATURE 19
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

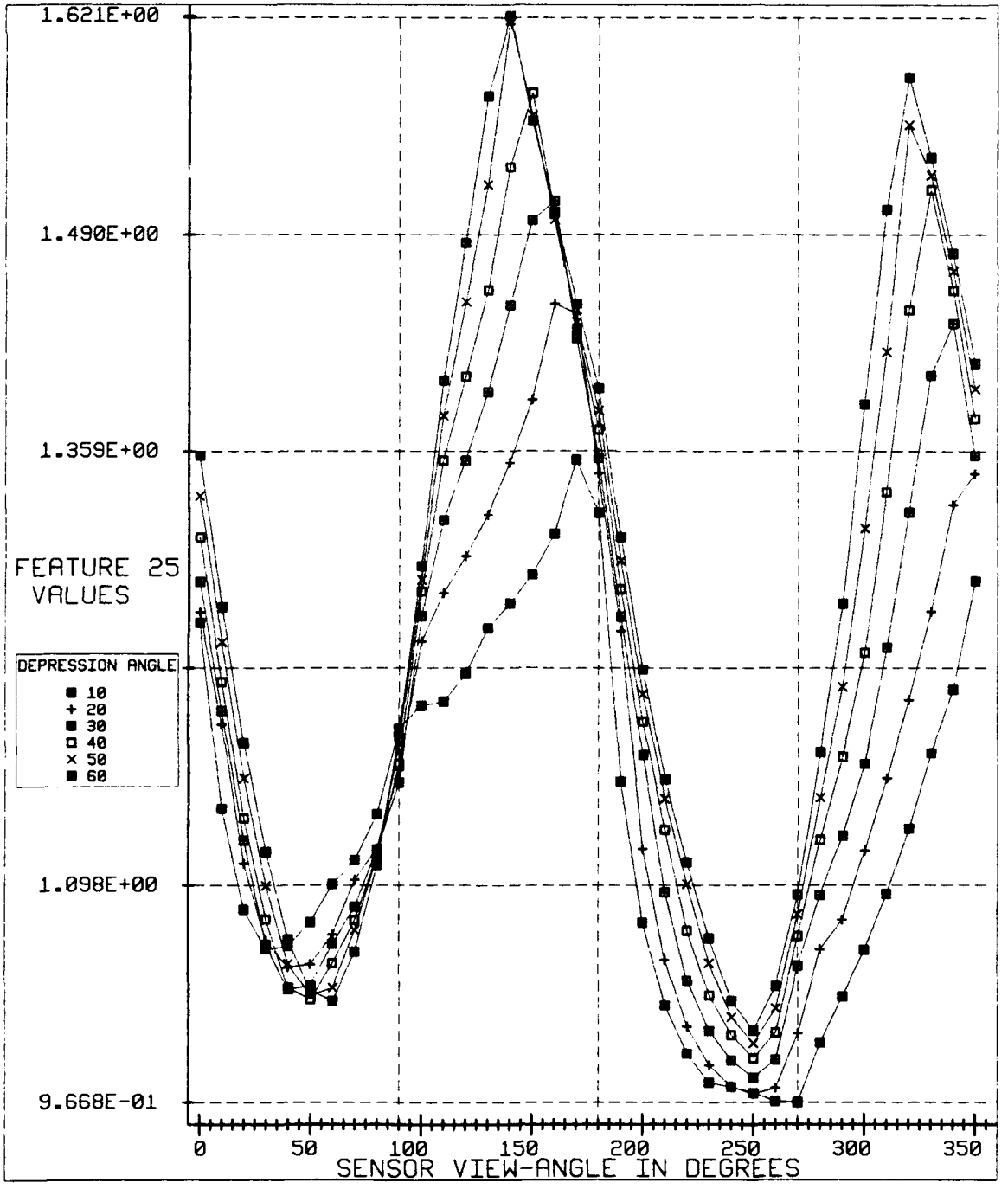

1.231E+00
1.184E+00
1.137E+00
1.043E+00
9.958E-01
FEATURE 20
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

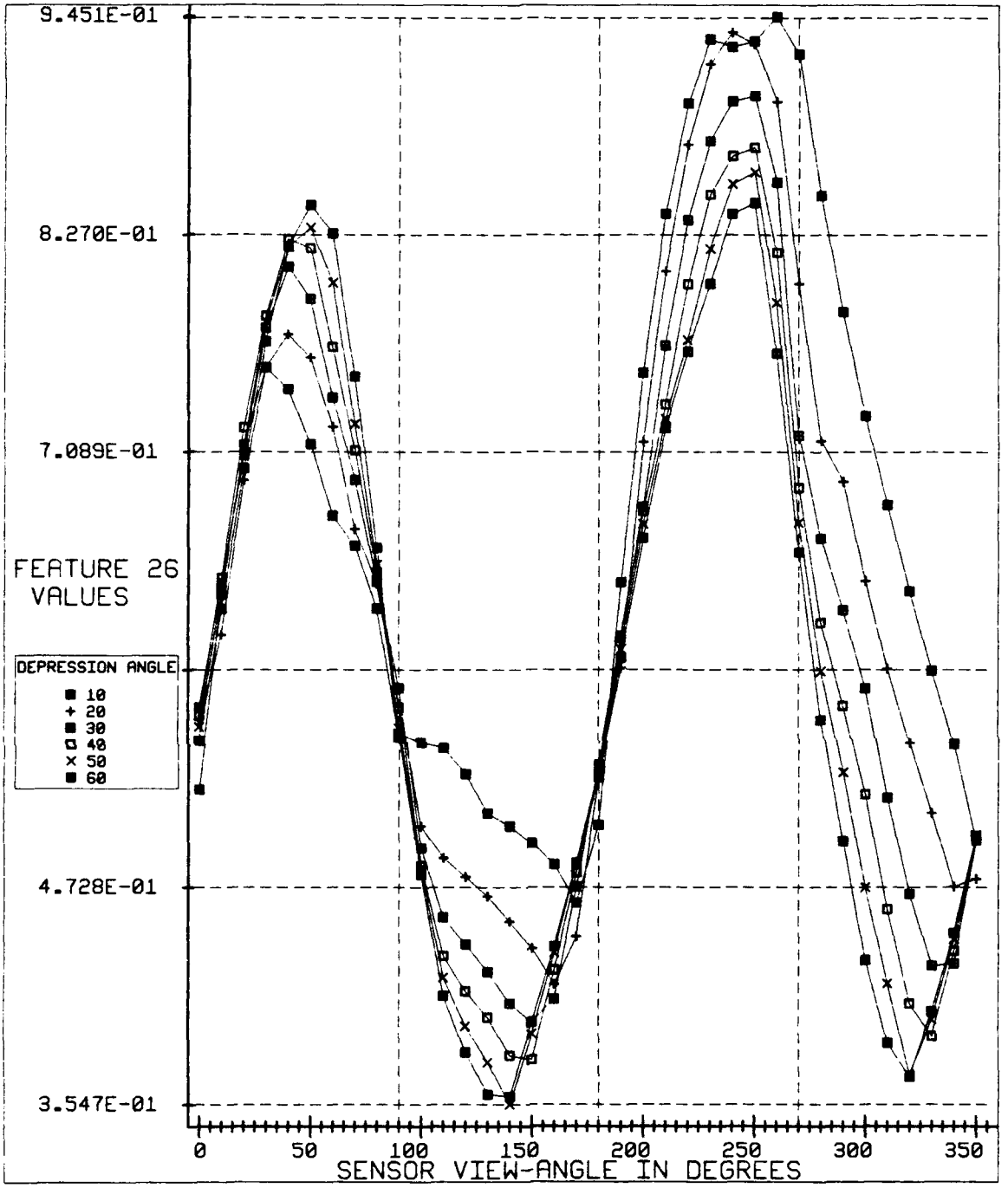

1.400E+00
1.330E+00
1.260E+00
1.121E+00
1.051E+00
FEATURE 21
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

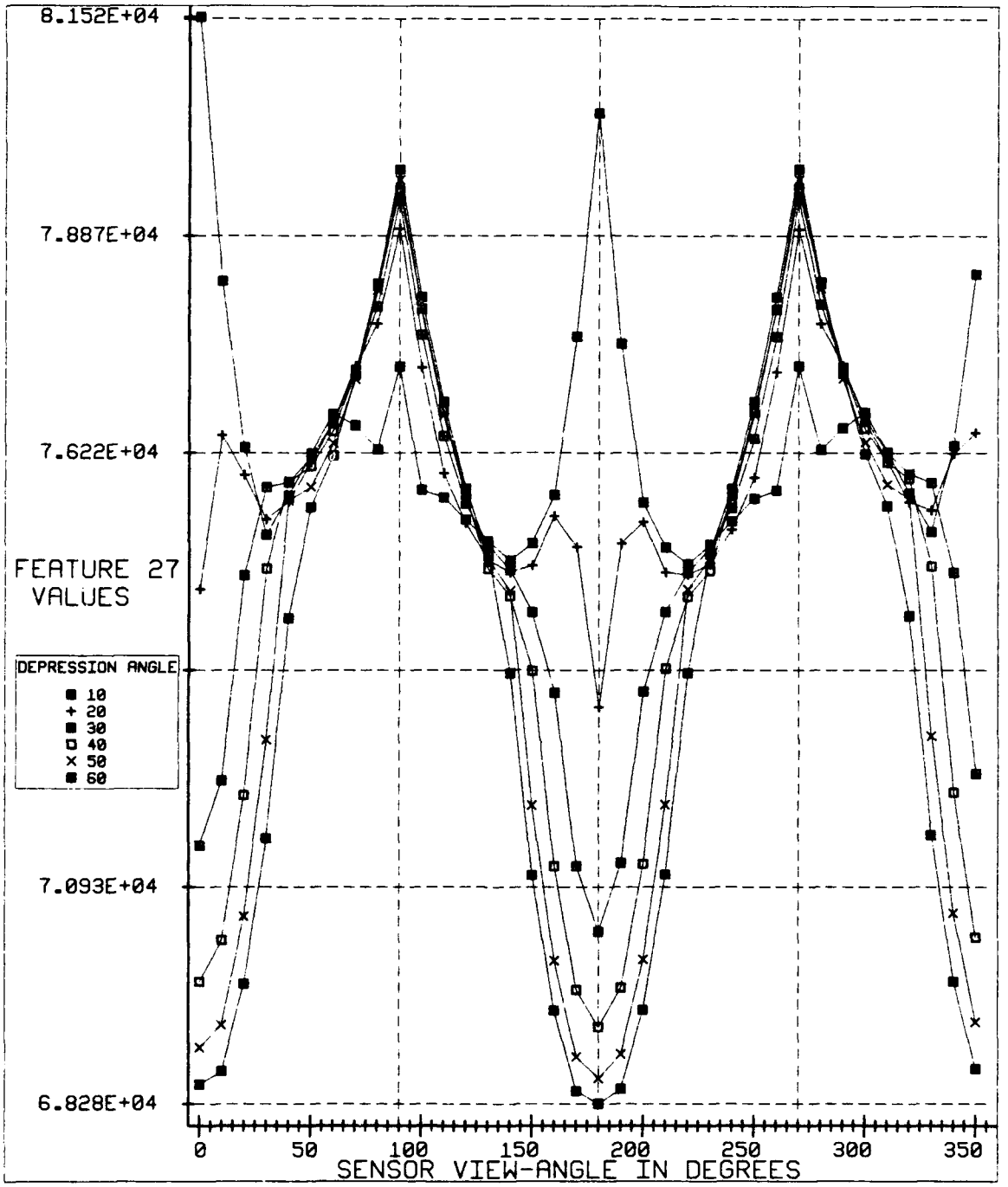

1.037E+00
9.836E-01
9.298E-01
8.221E-01
7.682E-01
FEATURE 22
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

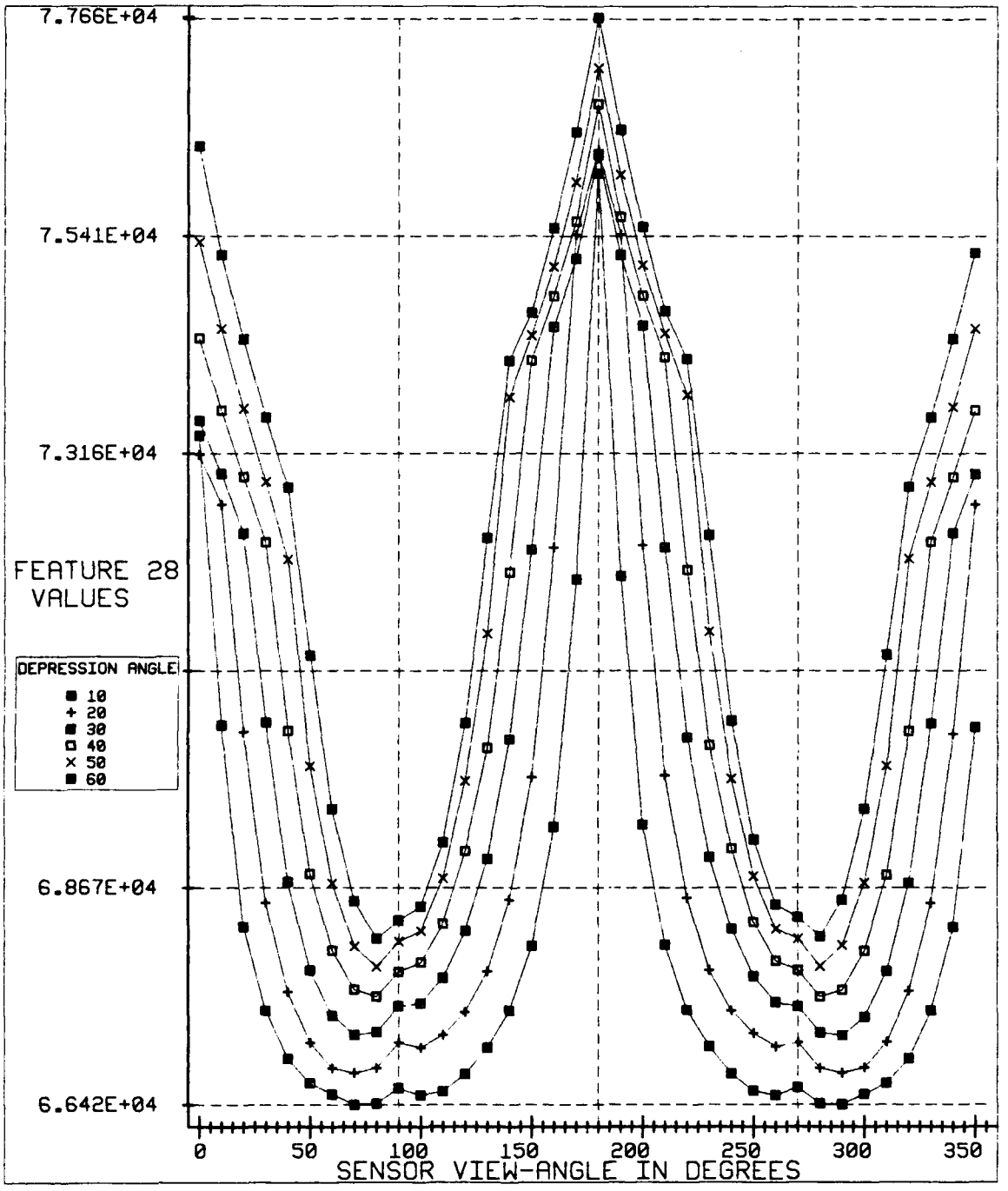

9.146E-01
8.449E-01
7.753E-01
6.361E-01
5.665E-01
FEATURE 23
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

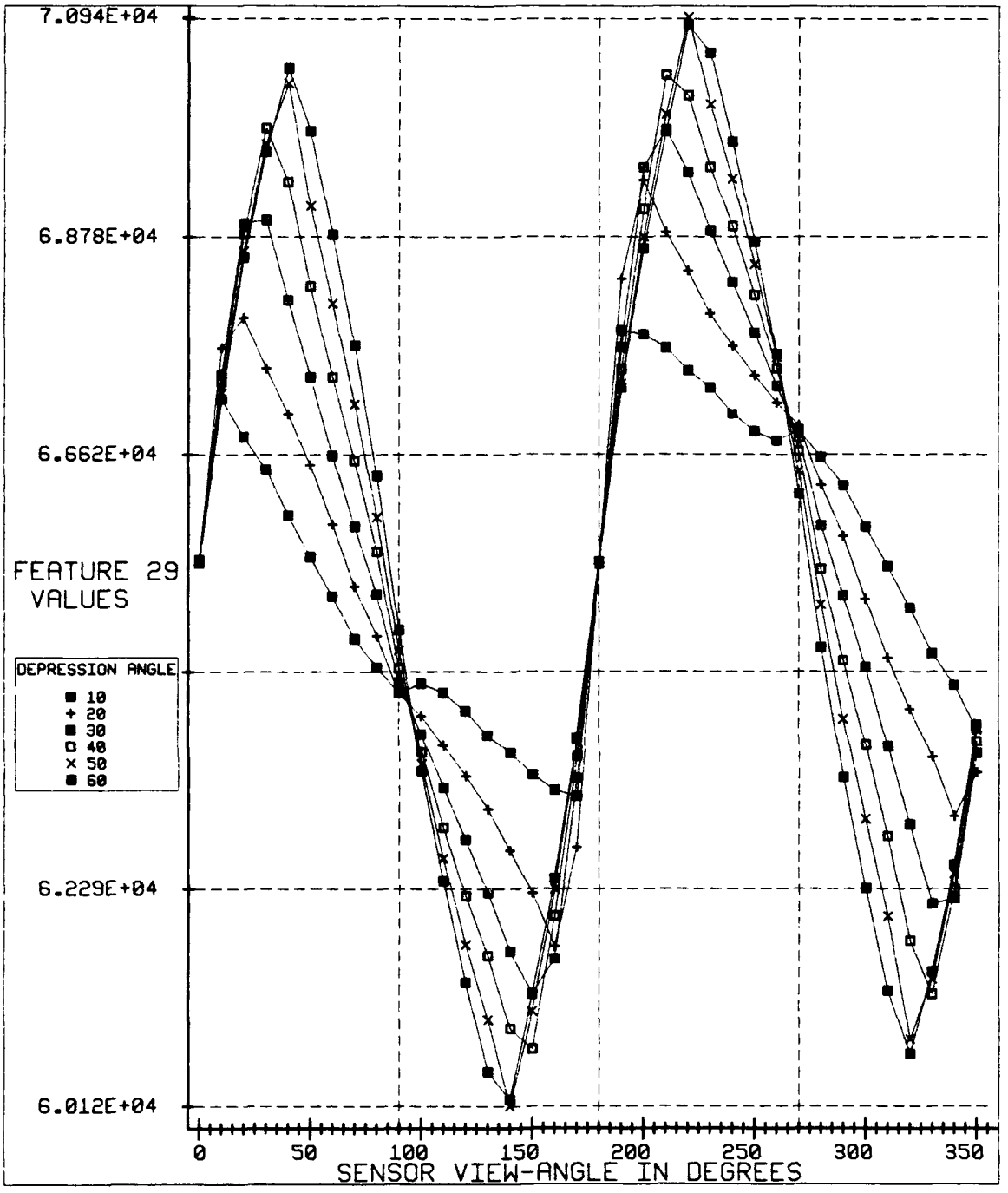

2.166E+00
1.951E+00
1.737E+00
FEATURE 24
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
1.308E+00
1.094E+00
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

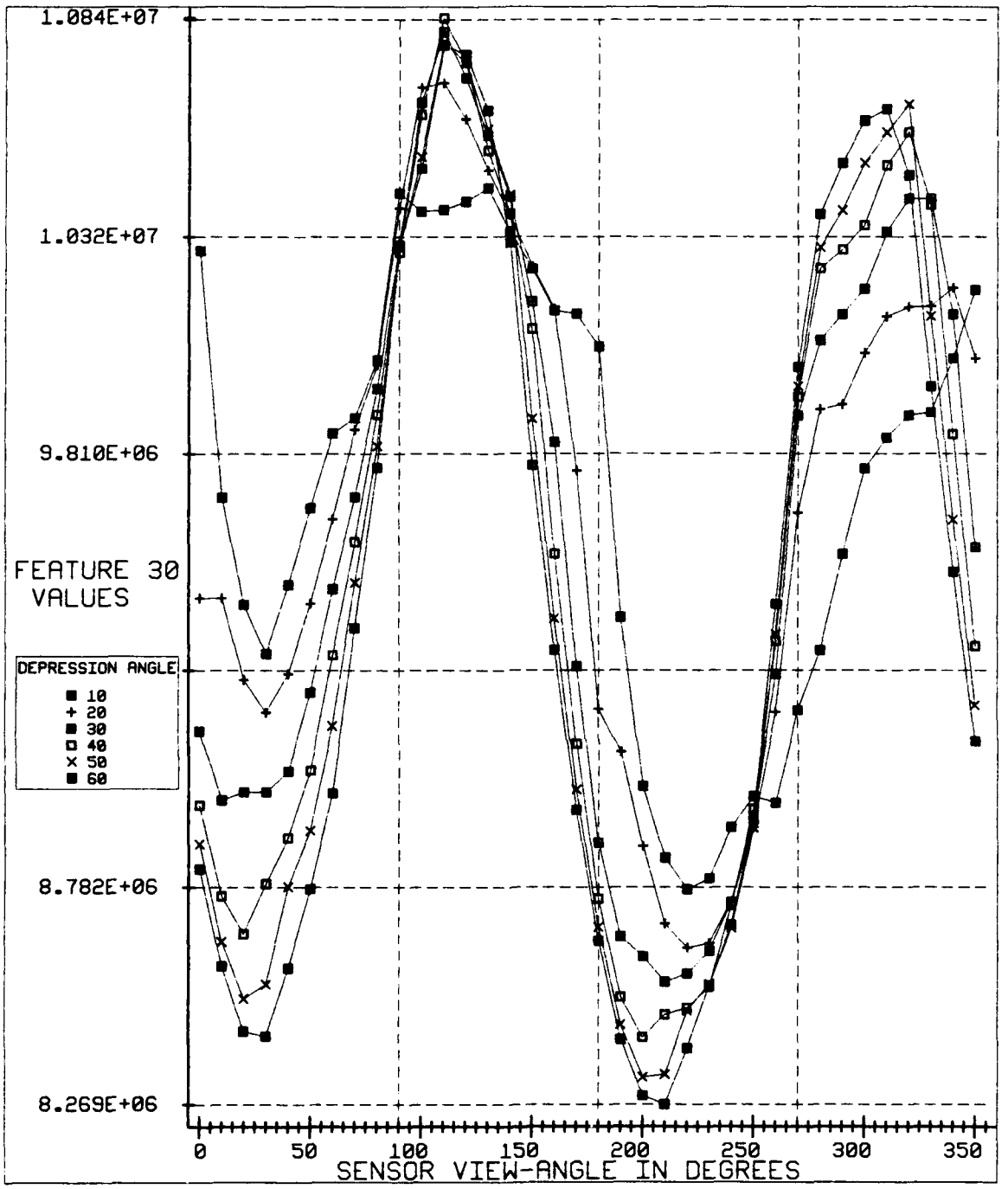

1.621E+00
1.490E+00
1.359E+00
1.098E+00
9.668E-01
FEATURE 25
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

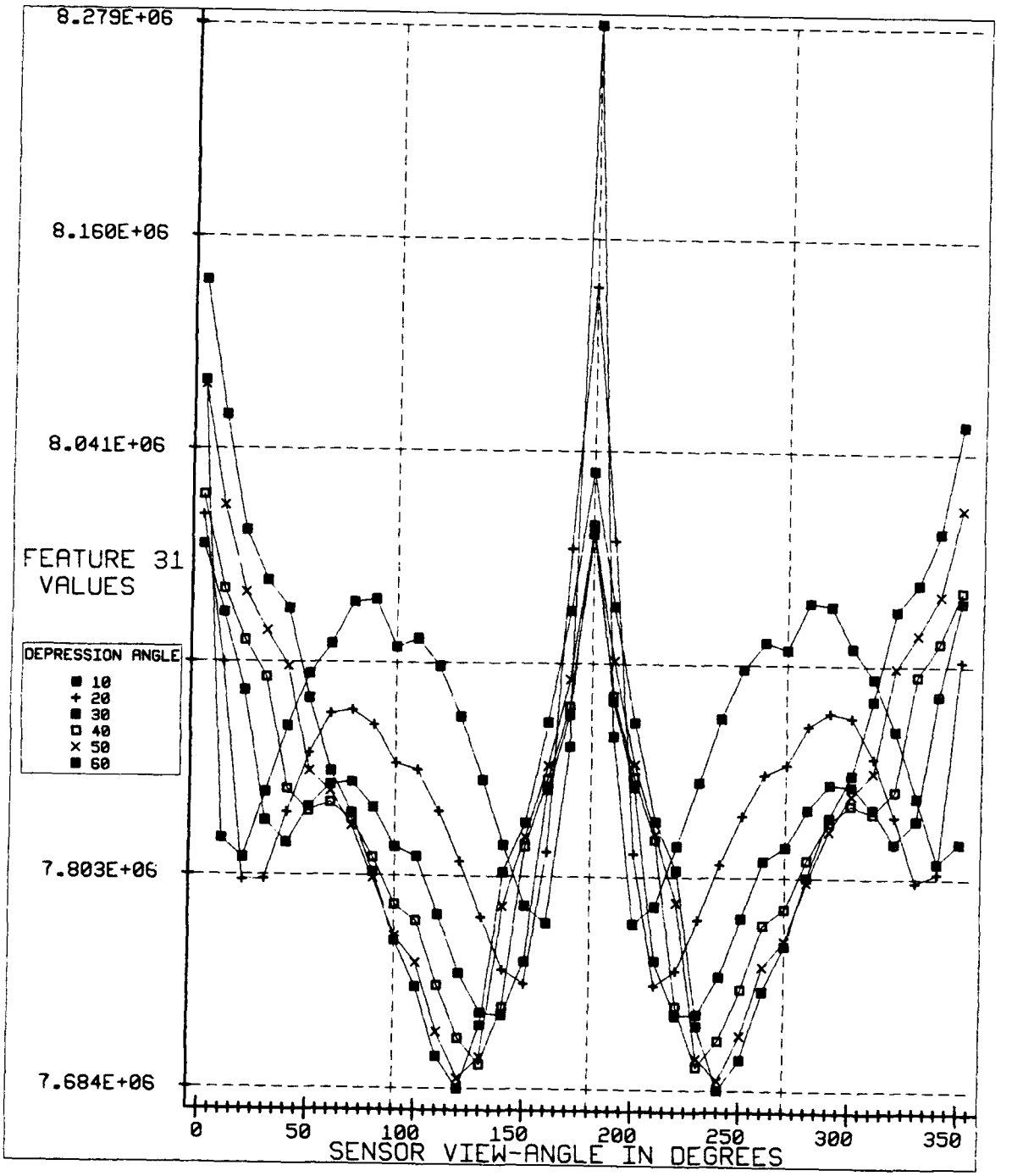

9.451E-01
8.270E-01
7.089E-01
4.728E-01
3.547E-01
FEATURE 26
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

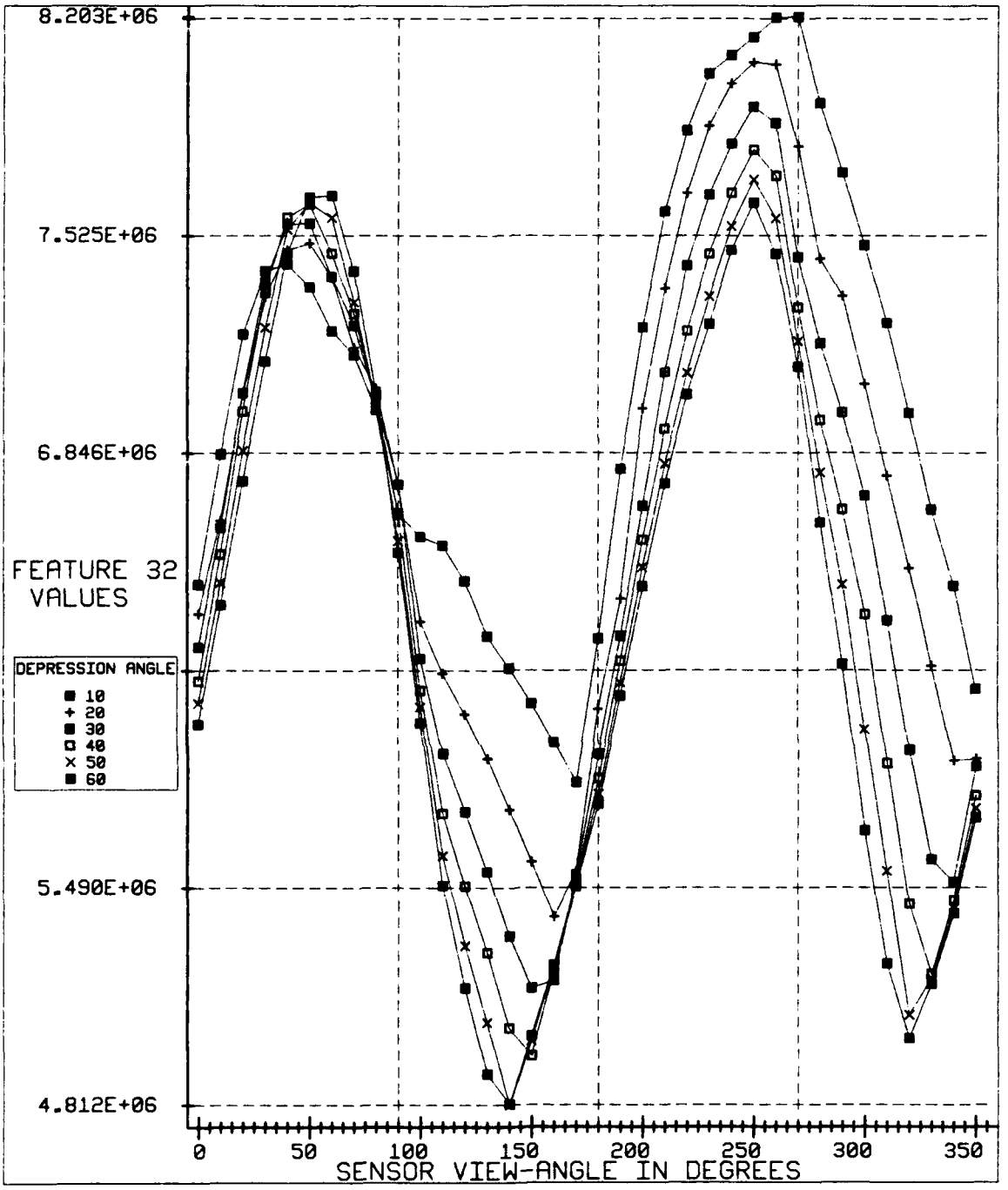

8.152E+04
7.887E+04
7.622E+04
FEATURE 27
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
7.093E+04
6.828E+04
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

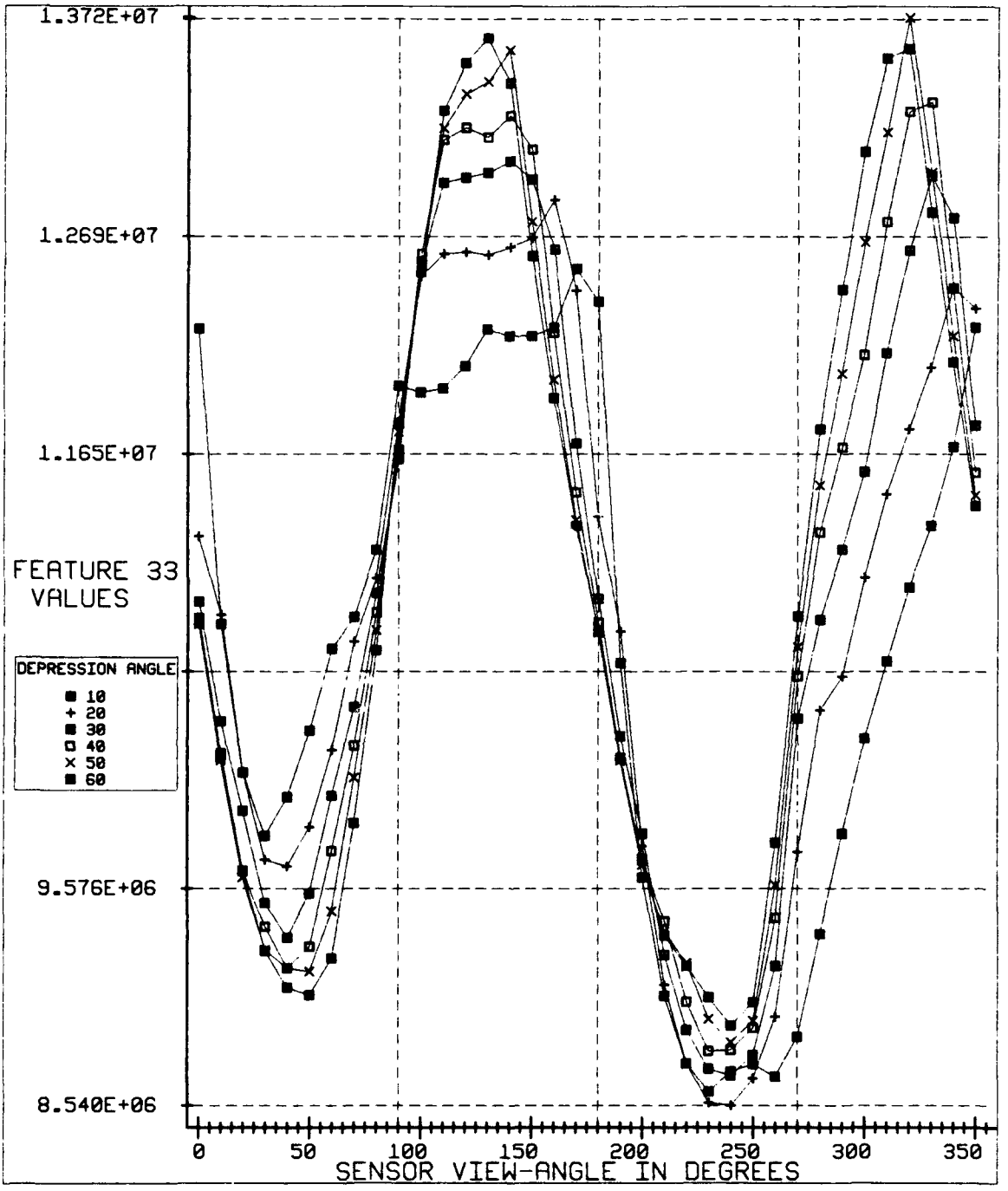

7.766E+04
7.541E+04
7.316E+04
6.867E+04
6.642E+04
FEATURE 28
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

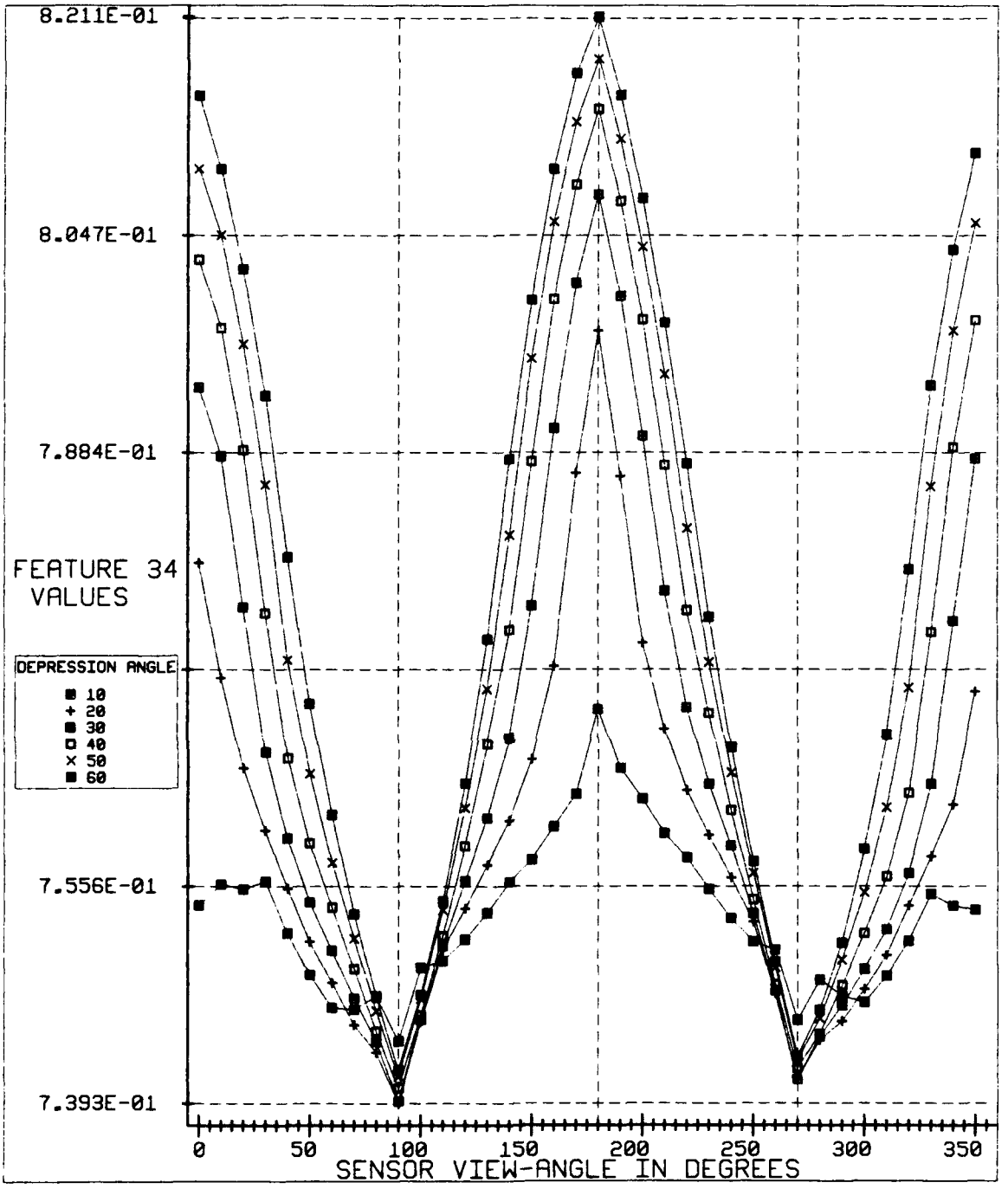

7.094E+04
6.878E+04
6.662E+04
6.229E+04
6.012E+04
FEATURE 29
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

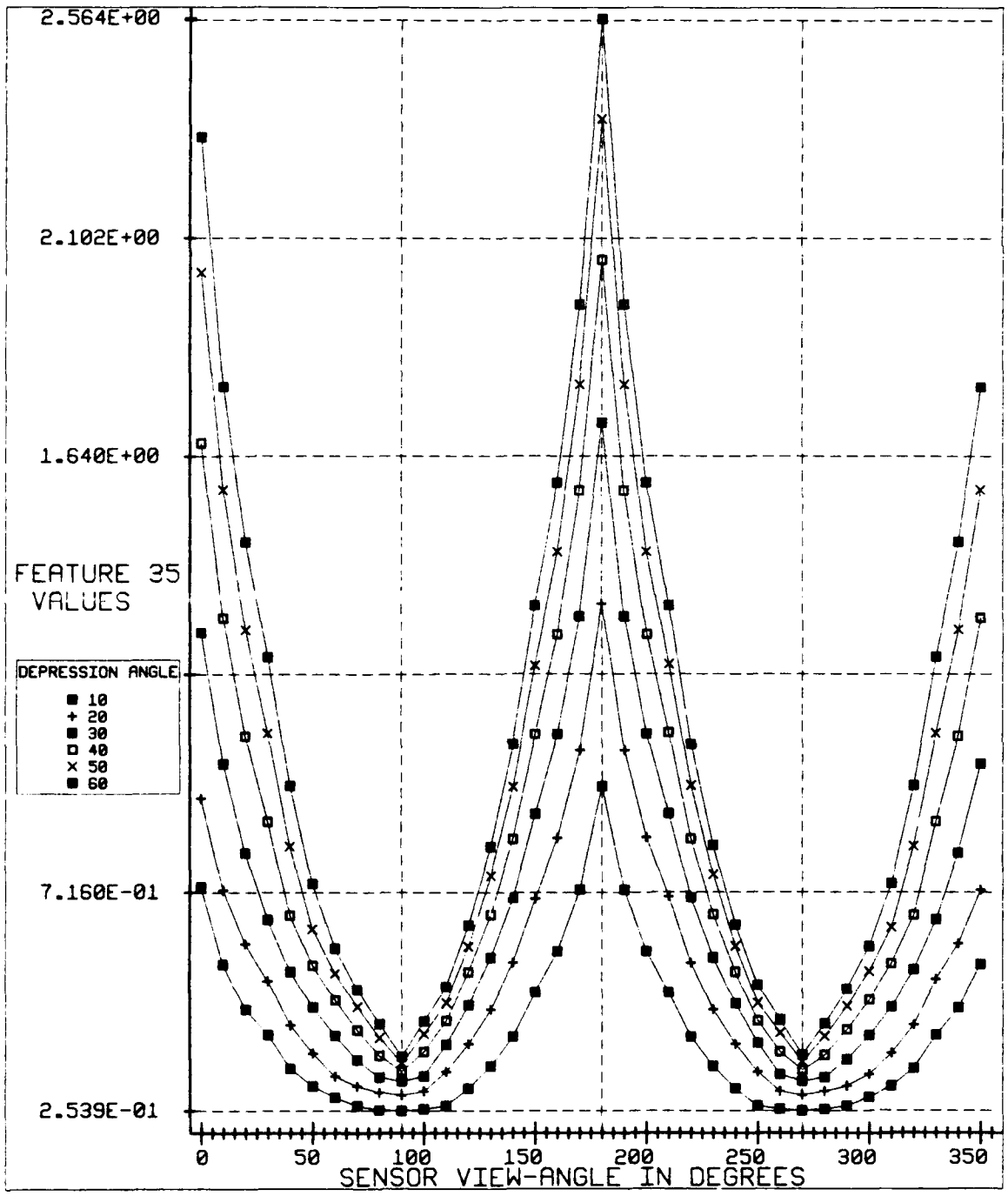

1.084E+07
1.032E+07
9.810E+06
8.782E+06
8.269E+06
FEATURE 30
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES


8.279E+06
8.160E+06
8.041E+06
FEATURE 31
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
7.803E+06
7.684E+06
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES


8.203E+06
7.525E+06
6.846E+06
5.490E+06
4.812E+06
FEATURE 32
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

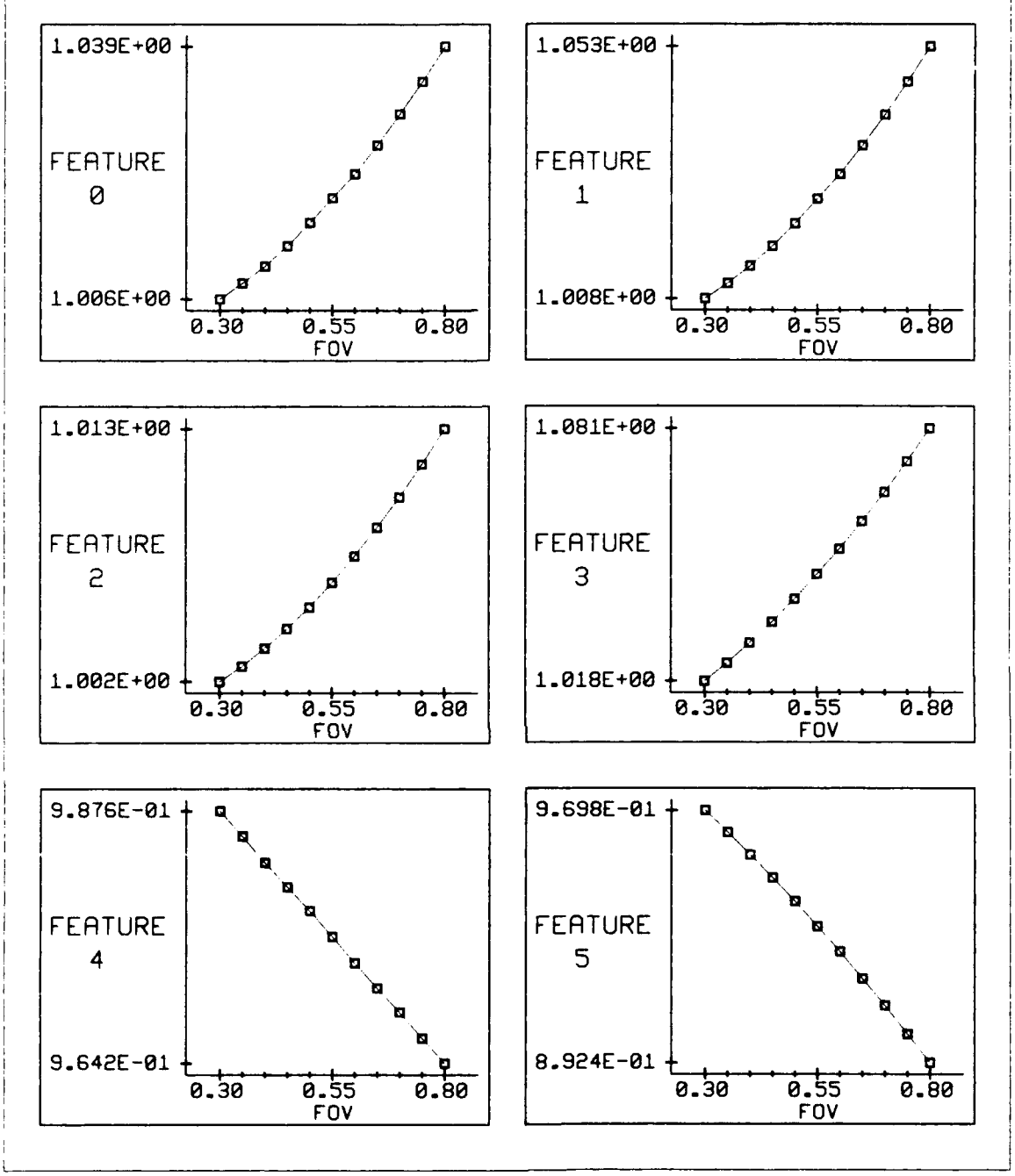

1.372E+07
1.269E+07
1.165E+07
9.576E+06
8.540E+06
FEATURE 33
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

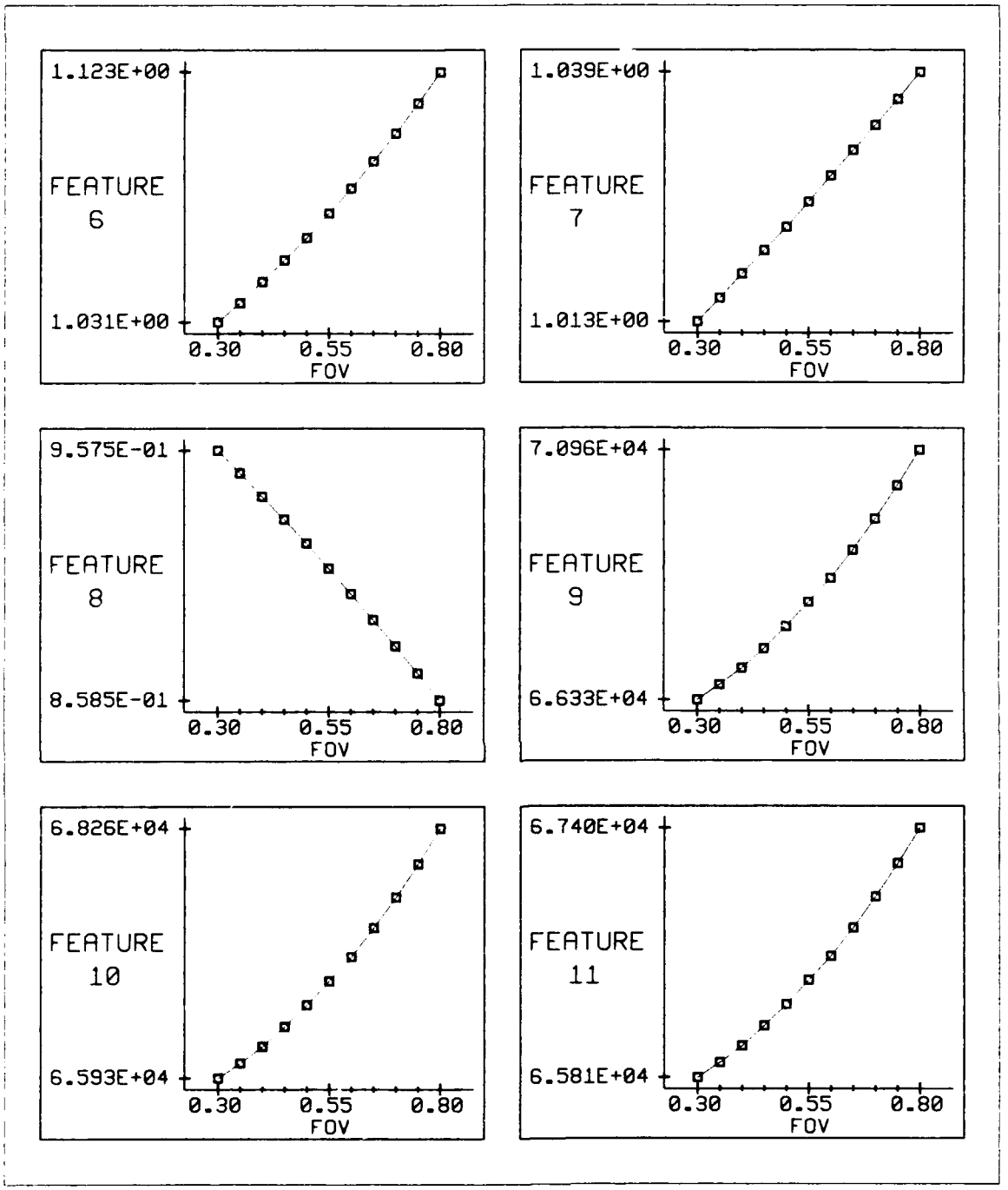

8.211E-01
8.047E-01
7.884E-01
7.556E-01
7.393E-01
FEATURE 34
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

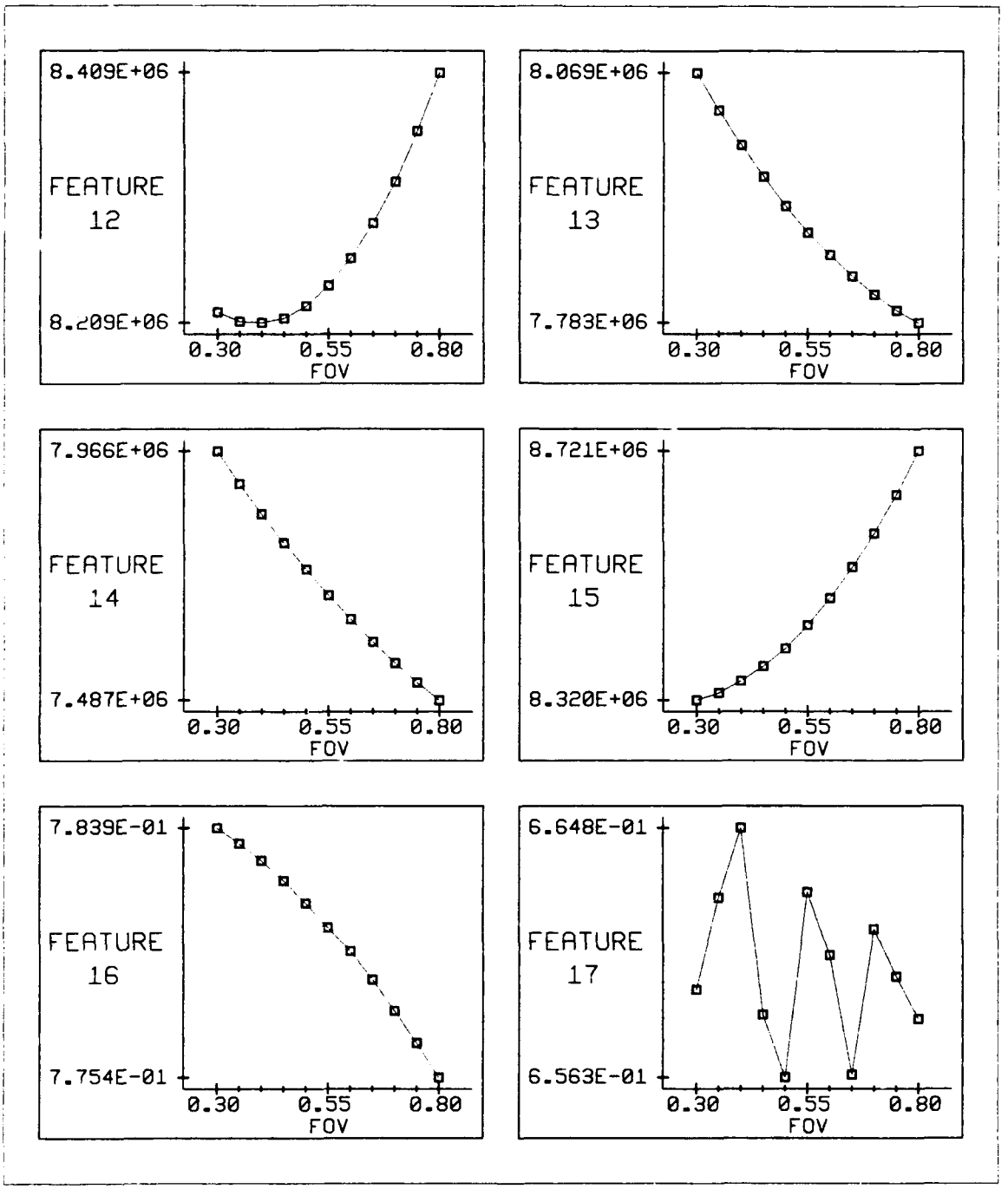

2.564E+00
2.102E+00
1.640E+00
FEATURE 35
VALUES
DEPRESSION ANGLE
10
20
30
40
50
60
7.160E-01
2.539E-01
0
50
100
150
200
250
300
350
SENSOR VIEW-ANGLE IN DEGREES

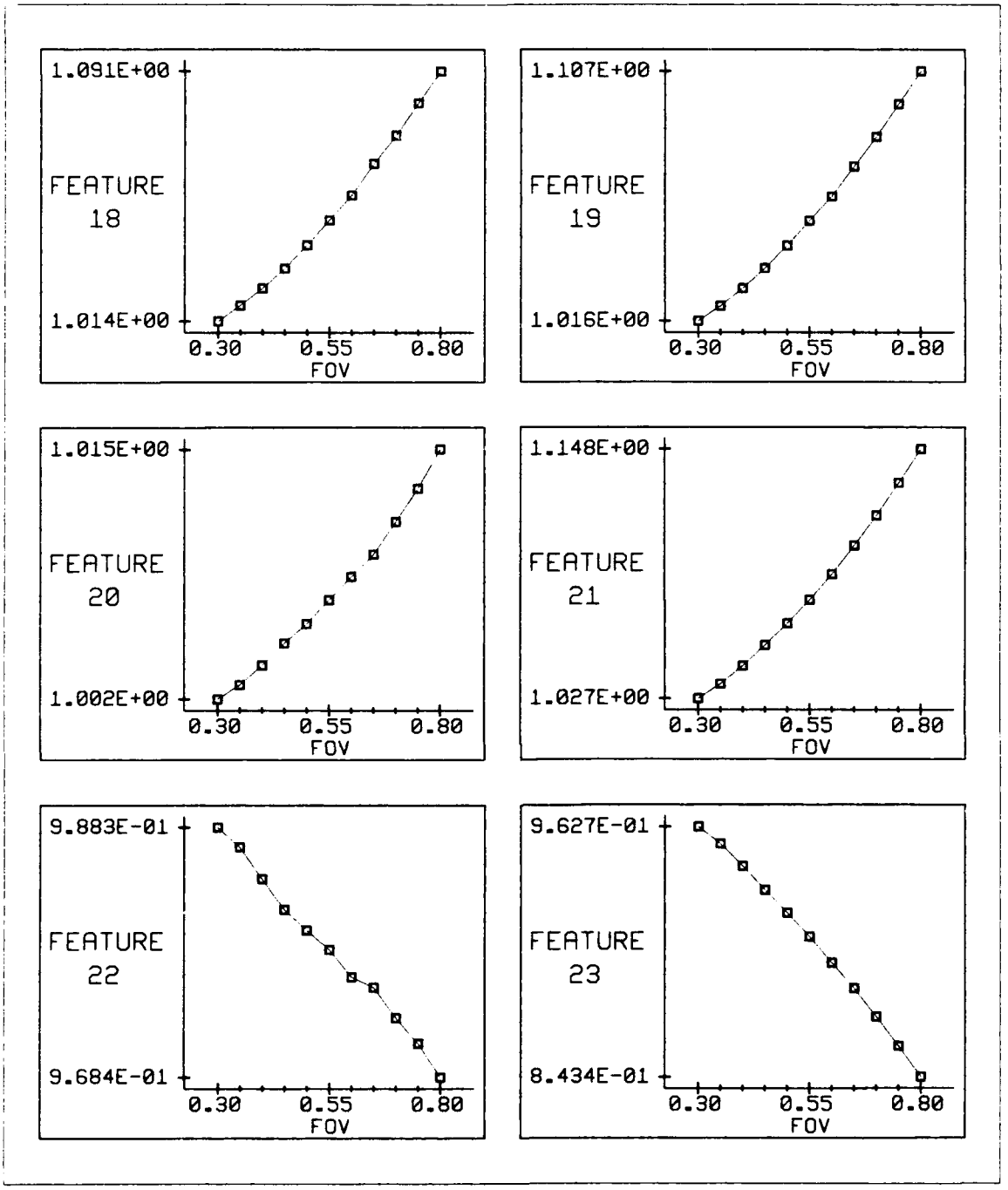
B.2 Scale Invariance Plots

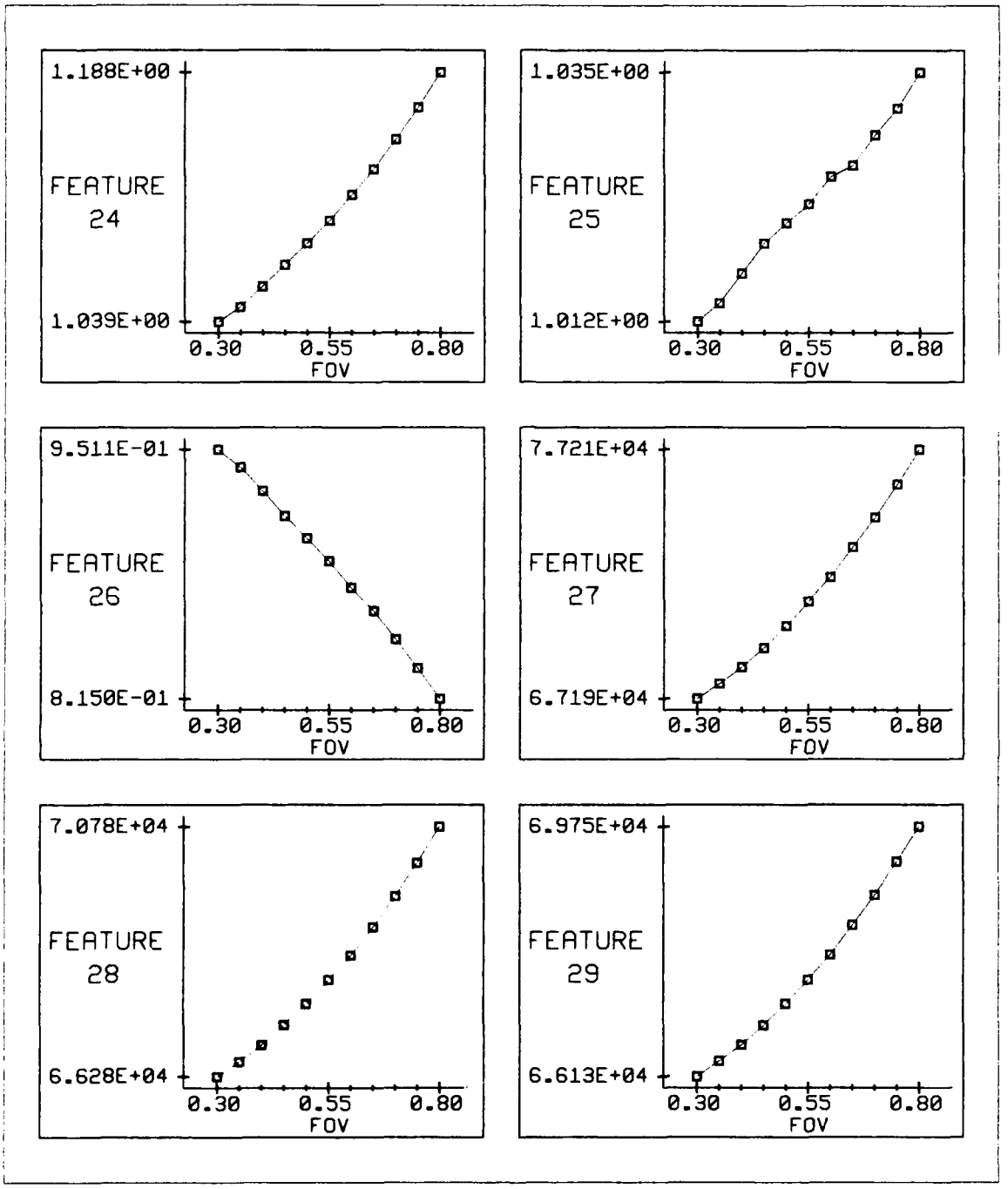
The following 36 graphs (six to a page) show how the features changed numerically as the object's FOV changed from 0.80 to 0.30.

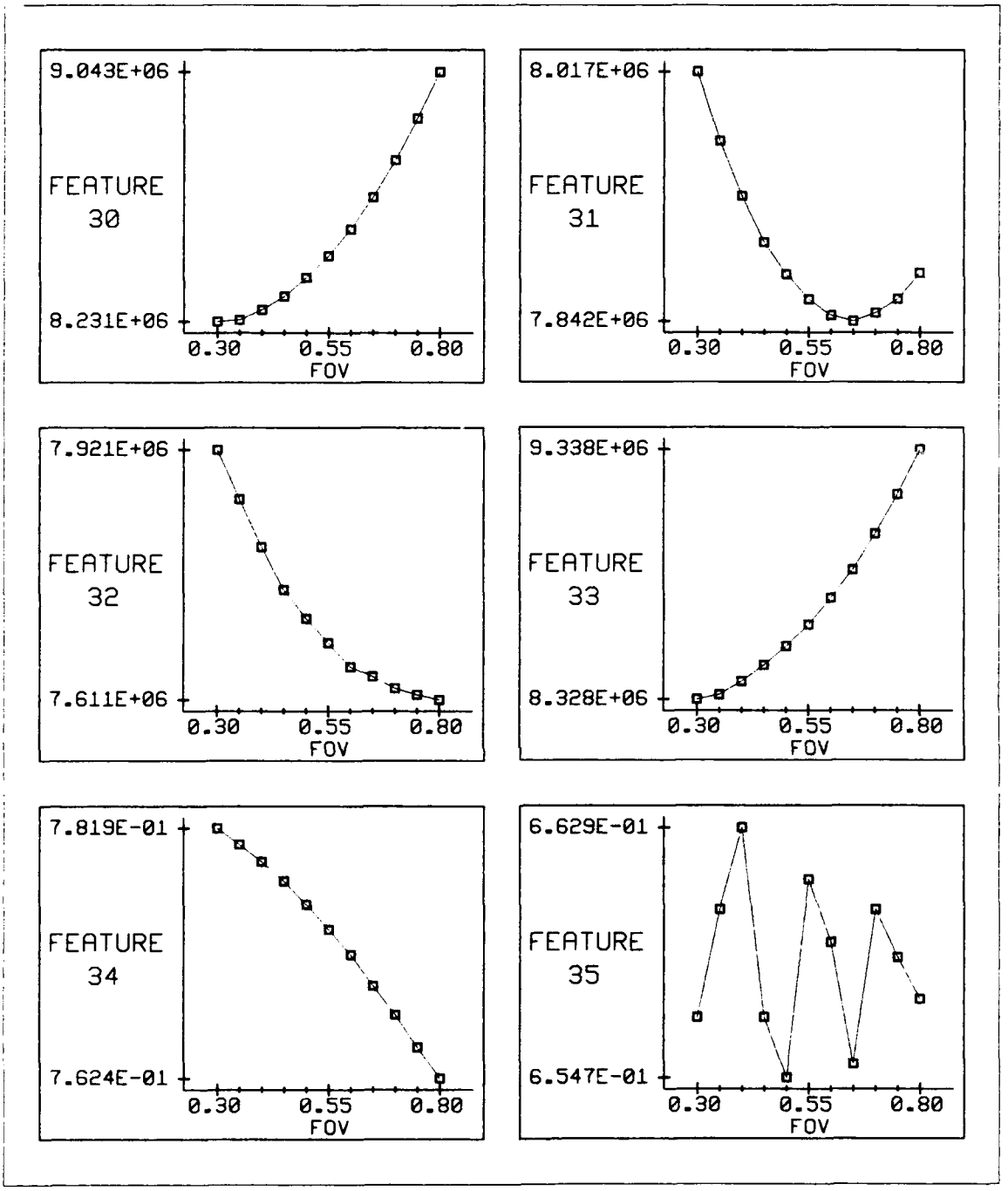

1.039E+00
FEATURE
0
1.006E+00
0.30
0.55
0.80
FOV
1.053E+00
FEATURE
1
1.008E+00
0.30
0.55
0.80
FOV
1.013E+00
FEATURE
2
1.002E+00
0.30
0.55
0.80
FOV
1.081E+00
FEATURE
3
1.018E+00
0.30
0.55
0.80
FOV
9.876E-01
FEATURE
4
9.642E-01
0.30
0.55
0.80
FOV
9.698E-01
FEATURE
5
8.924E-01
0.30
0.55
0.80
FOV


1.123E+00
FEATURE
6
1.031E+00
0.30
0.55
0.80
FOV
1.039E+00
FEATURE
7
1.013E+00
0.30
0.55
0.80
FOV
9.575E-01
FEATURE
8
8.585E-01
0.30
0.55
0.80
FOV
7.096E+04
FEATURE
9
6.633E+04
0.30
0.55
0.80
FOV
6.826E+04
FEATURE
10
6.593E+04
0.30
0.55
0.80
FOV
6.740E+04
FEATURE
11
6.581E+04
0.30
0.55
0.80
FOV


8.409E+06
FEATURE
12
8.209E+06
0.30
0.55
0.80
FOV
8.069E+06
FEATURE
13
7.783E+06
0.30
0.55
0.80
FOV
7.966E+06
FEATURE
14
7.487E+06
0.30
0.55
0.80
FOV
8.721E+06
FEATURE
15
8.320E+06
0.30
0.55
0.80
FOV
7.839E-01
FEATURE
16
7.754E-01
0.30
0.55
0.80
FOV
6.648E-01
FEATURE
17
6.563E-01
0.30
0.55
0.80
FOV


1.091E+00
FEATURE
18
1.014E+00
0.30
0.55
0.80
FOV
1.107E+00
FEATURE
19
1.016E+00
0.30
0.55
0.80
FOV
1.015E+00
FEATURE
20
1.002E+00
0.30
0.55
0.80
FOV
1.148E+00
FEATURE
21
1.027E+00
0.30
0.55
0.80
FOV
9.883E-01
FEATURE
22
9.684E-01
0.30
0.55
0.80
FOV
9.627E-01
FEATURE
23
8.434E-01
0.30
0.55
0.80
FOV


1.188E+00
FEATURE
24
1.039E+00
0.30
0.55
0.80
FOV
1.035E+00
FEATURE
25
1.012E+00
0.30
0.55
0.80
FOV
9.511E-01
FEATURE
26
8.150E-01
0.30
0.55
0.80
FOV
7.721E+04
FEATURE
27
6.719E+04
0.30
0.55
0.80
FOV
7.078E+04
FEATURE
28
6.628E+04
0.30
0.55
0.80
FOV
6.975E+04
FEATURE
29
6.613E+04
0.30
0.55
0.80
FOV


9.043E+06
FEATURE
30
8.231E+06
0.30
0.55
0.80
FOV
8.017E+06
FEATURE
31
7.842E+06
0.30
0.55
0.80
FOV
7.921E+06
FEATURE
32
7.611E+06
0.30
0.55
0.80
FOV
9.338E+06
FEATURE
33
8.328E+06
0.30
0.55
0.80
FOV
7.819E-01
FEATURE
34
7.624E-01
0.30
0.55
0.80
FOV
6.629E-01
FEATURE
35
6.547E-01
0.30
0.55
0.80
FOV

Bibliography

1. Abidi, M.A. and Chandra, T. "Pose Estimation for Camera Calibration and Landmark Tracking," *Proceedings of the 1990 IEEE International Conference on Robotics and Automation.* 420-426. Cincinnati, OH. May 1990.
2. Abidi, M.A. and Chandra, T. "Evaluation of a Pose Estimation Algorithm Using Single Perspective View," *Proceedings of the SPIE - The International Society for Optical Engineering, v1382.* 409-426. Bellingham, WA, 1990.
3. Abu-Mostafa, Yaser S. and Psaltis, Demetri "Recognitive Aspects of Moment Invariants," *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-6/6* :698-706 (Nov 1984).
4. Akerman, Alexander III *et al* "Improved Delta-t and Size Metrics for Thermally Structured Targets," *Proceedings of the SPIE - The International Society for Optical Engineering, v1310.* 95-99. Bellingham, WA, 1990.
5. Arbter, Klaus *et al.* "Application of Affine-Invariant Fourier Descriptors to Recognition of 3-D Objects," *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-12/7* :640-647. (Jul 1990).
6. Ayer, Capt Kevin W. *Gabor Transforms for Forward Looking Infrared Image Segmentation.* MS thesis, AFIT/GEO/ENG/89D-01. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December 1989.
7. Ballard, D.H. "Generalizing the Hough Transform to Detect Arbitrary Shapes," *Pattern Recognition, 12/2* :111-122 (1981).
8. Bart, Mischa *et al.* "A Learning Procedure for the Recognition of 3D Objects from 2D Images," *Proceedings of the SPIE - The International Society for Optical Engineering, v1381.* 66-77. Bellingham, WA, 1990.
9. Bell, Benjamin and Pau, L. F. "Contour Tracking and Corner Detection in a Logic Programming Environment," *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-12/9* :913-917 (Sept 1990).
10. Besl, Paul J. and Jain, Ramesh C. "Segmentation Through Variable-Order Surface Fitting," *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-10/2* 167-192 (Mar 1988).
11. Bidlack, Clint and Trivedi, Mohan "An Integrated Vision System for Object Identification and Localization Using Three-Dimensional Geometric Models," *Proceedings of the SPIE - The International Society for Optical Engineering, v1468.* 270-280. Bellingham, WA, 1991.
12. Bolles, R.C. and Cain, R.A. "Recognizing and Locating Partially Visible Objects: The Local-Feature-Focus Model," *International Journal of Robotics Research, v1/n3* :57-82 (1982).

13. Brooks, Rodney "Symbolic Reasoning Among 3-D Models and 2-D Images," *Artificial Intelligence, 17* :285-348 (1981).

14. Brooks, Rodney "Model-Based Three-Dimensional Interpretations of Two-Dimensional Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-5/2* :140-150. (Mar 1983).

15. Casasent, David and Liebowitz, Suzanne "Model-Based Knowledge-Based Optical Processors," *Applied Optics, 26/10* :1935-1942. (May 1987).

16. Casasent, David and Krishnapuram, Raghu "Determination of Three-Dimensional Object Location and Orientation from Range Images," *IEEE Transaction on Pattern Analysis and Machine Intelligence, PAMI-11/11* :1158-1167 (Nov 1989).

17. Chatravarty, Indranil and Freeman, Herbert "Characteristic Views as a Basis for Three-Dimensional Object Recognition," *Proceedings of the SPIE - The International Society for Optical Engineering, v336.* 37-45. Bellingham, WA, 1982.

18. Chen, Nai-Yung *et al.* "Estimating Workpiece Pose Using the Feature Points Method," *IEEE Transactions on Automatic Control, AC-25* :1027-1041. (Dec 1980).

19. Chen, Zen and Ho, Shinn-Ying "Computer Vision for Robust 3D Aircraft Recognition with Fast Library Search," *Pattern Recognition, 24/5* :375-390. (1991).

20. Chien, C.H. and Aggarwal, J.K. "Volume/Surface Octrees for the Representation of Three-Dimensional Objects," *Computer Vision, Graphics, and Image Processing, 36* :100-113 (1986).

21. Cyganski, David *et al.* "A Tensor Operator Method for Identifying the Affine Transformation Relating Image Pairs," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition.* 361-363. Washington, D.C., Jun 1983.

22. Cyganski, D. and Orr, J. "Object Identification and Orientation Determination in 3-Space with No Point Correspondence Information," *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, v2.* p23.8.1-23.8.4 San Diego, CA, Mar 1984.

23. Daugman, John G. "Uncertainty Relation for Resolution in Space, Spatial Frequency, and Orientation Optimized by Two-Dimensional Visual Cortex Filters," *J. Opt. Soc. Am, 2/7* :1160-1169 (Jul, 1985).

24. Daugman, John G. "Complete Discrete 2-D Gabor Transforms by Neural Networks for Image Analysis and Compression," *IEEE Transactions on Acoustics, Speech, and Signal Processing, 36/7* :1169-1179 (Jul 1988).

25. DiMarco, John *et al.* "Development of the Modular Infra-Red Target Imaging Model (MIRTIM)," *Proceedings of the SPIE - The International Society for Optical Engineering, v1050.* 120-127. Bellingham, WA, 1989.

26. Johnson, Keith and Rodriguez, Leonard *User's Manual for TCM2.* Georgia Tech Research Institute. Atlanta: Georgia Institute of Technology, 1990.

27. Duda, Richard O. *et al.* "Use of Range and Reflectance Data to Find Planar Surface Regions," *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-1/3* :259-271 (Jul 1979).

28. Dudani, Sahibsingh A. *et al.* "Aircraft Identification by Moment Invariants," *IEEE Transactions on Computers, C-26/1* :39-45 (Jan 1977).

29. Fan, T.-J.; Nevatia, R.; Medioni, G. "Recognizing 3-D Objects Using Surface Descriptions," *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-11/11* :1140-1157 (Nov 1989).

30. Fischer, Martin and Bolles, Robert "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography," *Communications of the ACM, 24/6* :381-395 (Jun 1981).

31. Flynn, Patrick J. and Jain, Anil K. "CAD-Based Computer Vision: From CAD Models to Relational Graphs," *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-13/2* :114-132 (Feb 1991).

32. Gaskill, Jack D. *Linear Systems, Fourier Transforms, and Optics.* New York: John Wiley & Sons, 1978.

33. Gennery, D.B. "Object Detection and Measurement Using Stereo Vision," *Proceedings of the Sixth International Joint Conference on Artificial Intelligence* 320-327. Tokyo, Japan. Aug 1979.

34. Gilmore, John "Knowledge-Based Target Recognition System Evolution," *Optical Engineering, 30/5* :557-570. (May 1991).

35. Gizzi, Martin S. *et al.* "Selectivity for Orientation and Direction of Motion of Single Neurons in Cat Striate and Extrastriate Visual Cortex," *Journal of Neurophysiology, 63/6* :1529-1543 (Jun 1990).

36. Goodman, Joseph W. *Fourier Optics.* New York: McGraw-Hill Publishing Company, 1968.

37. Grimson, W.E.L. and Huttenlocher, D.P. "On the Sensitivity of the Hough Transform for Object Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-12/3* :255-274 (Mar 1990).

38. Grimson, W.E.L. and Lozano-Perez, T. "Model-Based Recognition and Localization from Sparse Range or Tactile Data," *International Journal of Robotics Research, v3/n3* :3-35 (Fall 1984).

39. Grimson, W.E.L. and Lozano-Perez, T. "Localizing Overlapping Parts by Searching the Interpretation Tree," *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-9/4* :469-4°2 (Jul 1987).

40. Haralick, Robert "Determining Camera Parameters from the Perspective Projection of a Rectangle," *Pattern Recognition, 22/3* :225-230 (1989).

41. Haralick, Robert and Joo, Hyonam "2D-3D Pose Estimation," *Ninth International Conference on Pattern Recognition.* 385-400. Rome, Italy, Nov 1988.

42. Horn, Berthold "Extended Gaussian Images," *Proceedings of the IEEE, 72/12.* 1671-1686. Dec 1984.

43. Horn, Berthold *Robot Vision.* Cambridge, Massachusetts: The MIT Press, 1986.

44. Hu, M.K. "Visual Pattern Recognition by Moment Invariants," *IRE Transactions on Information Theory, vol. IT-8* :179-187 (1962).

45. Hubel, David H. and Wiesel, Torsten N. "Brain Mechanisms of Vision," *Scientific American, 241/3* :179-187 (1979).

46. Ikeuchi, K. "Recognition of 3-D Objects Using the Extended Gaussian Image," *Proceedings of the Seventh International Joint Conference on Artificial Intelligence* 595-600. Vancouver, B.C., Canada. 1981.

47. Korn, Matthew and Dyer, Charles "3-D Multiview Object Representation for Model-Based Object Recognition," *Pattern Recognition, 20/1* :91-103 (1987).

48. Kriegman, David and Ponce, Jean "On Recognizing and Positioning Curved 3-D Objects from Image Contours," *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-12/12* :1127 (Dec 1990).

49. Le, Capt Phung D. *Model-Based 3-D Recognition System Using Gabor Features and Neural Networks.* MS thesis, AFIT/GE/ENG/90D-05. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December 1990.

50. Levine, M.D. "Feature Extraction: A Survey," *Proceedings of the IEEE, 57.* 1391-1407. Aug 1969.

51. Lie, Wen-Nung *et al.* "Model-Based Recognition and Positioning of Polyhedra Using Intensity-Guided Range Sensing and Interpretation in 3-D Space," *Pattern Recognition, 23/9* :983-997 (1990).

52. Lindahl, Charles *et al.* "Synthetic, Multisensor Database Generation and Validation," *Proceedings of the SPIE - The International Society for Optical Engineering, v1310.* 88. Bellingham, WA, 1990.

53. Lucero, A.B. *et al.* "Classifiability of Infrared Target Signatures," *Proceedings of the SPIE - The International Society for Optical Engineering, v1050.* 50-58. Bellingham, WA, 1989.

54. Lowe, David "Three-Dimensional Object Recognition from Single Two-Dimensional Images," *Artificial Intelligence, 31* :355-395 (1987).

55. Martin, Worthy and Aggarwal, J.K. "Survey: Dynamic Scene Analysis," *Computer Graphics and Image Processing, 7* :356-374 (Jun 1978).

56. Martin, Worthy and Aggarwal, J.K. "Volumetric Descriptions of Objects from Multiple Views," *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-5/2* :150-158 (Mar 1983).

57. Mohan, Rakesh and Nevatia, Ramakant "Using Perceptual Organization to Extract 3-D Structures," *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-11/11* :1121-1139 (Nov 1989).

58. Nagel, H.-H. "Image Sequence Analysis: What Can We Learn from Applications," *Image Sequence Analysis*, T.S. Huang, Ed. Heidelberg, Germany: Springer-Verlag :19-228 (Mar 1981).

59. Oshima, M. and Shirai, Y. "A Scene Description Method Using Three-Dimensional Information," *Pattern Recognition, 11* :9-17 (1979).

60. Oshima, M. and Shirai, Y. "Object Recognition Using Three-Dimensional Information," *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-3/4* :353-361 (Jul 1983).

61. Pavlidis, Theodosios "Fuzzy Representations as Means of Overcoming the Overcommitment of Segmentation," *IEEE Conference on Computer Graphics, Pattern Recognition, and Data Structure* 215-219. Los Angeles, CA. May 1975.

62. Pentland, Alex P. "Local Shading Analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-6/2* :170-187 (Mar 1984).

63. Politopoulos, Anastasios "Minimum Edge Distance Classifier," *Proceedings of the SPIE - The International Society for Optical Engineering, v1050.* 59-71. Bellingham, WA, 1989.

64. Pong, Ting-Chuen *et al.* "Shape from Shading Using the Facet Model," *Pattern Recognition, 22/6* :683-695 (1989).

65. Ray, L.P. "Estimation of Modeled Object Pose from Monocular Images," *Proceedings of the 1990 IEEE International Conference on Robotics and Automation.* 408-413. Cincinnati, OH. May 1990.

66. Richard, Charles W., Jr and Hemani, Hooshang "Identification of Three-Dimensional Objects Using Fourier Descriptors of the Boundary Curve," *IEEE Transactions on Systems, Man, and Cybernetics, SMC-4/4* :371-378 (Jul 1974).

67. Roberts, G. *et al.* "Performance Measurements of Real-Time Feature Extraction Modules," *Proceedings of the SPIE - The International Society for Optical Engineering, v1310.* 32-45. Bellingham, WA. 1990.

68. Roggeman, Capt Michael C. *Multiple Sensor Fusion for Detecting Targets in FLIR and Range Images.* PhD dissertation. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, May 1989.

69. Roggeman, Michael C.; Rogers, Steven K.; *et al.* "Segmentation of Noisy Range Images Using the Small-Scale Planarity of Man-Made Vehicles," *Optical Engineering, 30/4* :408-413 (Apr 1991).

70. Rosenfeld, Azriel and Lee, Chia-Hong "Improved Methods of Estimating Shape from Shading Using the Light Source Coordinate System," *Artificial Intelligence, 26* :125-143 (1985).

71. Ruck, Capt Dennis W. *Multisensor Target Detection and Classification.* MS thesis, AFIT/GE/ENG/87D-56. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December 1987.

72. Rushmeier, Holly and Tynor, Stephen "Incorporating the BRDF into an Infrared Scene Generation System," *Proceedings of the SPIE - The International Society for Optical Engineering, v1311.* 109-118. Bellingham, WA, 1990.

73. Sadjadi, Firooz A. and Bazados, Mike "A Perspective on ATR Evaluation Technology," *Proceedings of the SPIE - The International Society for Optical Engineering, v1310.* 2-15. Bellingham, WA, 1990.

74. Sadjadi, Firooz A. and Hall, Ernest L. "Three-Dimensional Moment Invariants," *IEEE Transaction on Pattern Analysis and Machine Intelligence, PAMI-2/2* :127-136 (Mar 1980).

75. Silberberg, T.M. *et al.* "An Iterative Hough Procedure for Three-Dimensional Object Recognition," *Pattern Recognition, 17* :621-628 (1984).

76. Stockman, George *et al.* "Matching Images to Models for Registration and Object Detection via Clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-4/3* :229-241 (May 1982).

77. Stockman, George and Flinchbaugh, Bruce "Recognition via Alignment Using Aspect Models," *Proceedings of the SPIE - The International Society for Optical Engineering, v1293.* 224-235. Bellingham, WA, 1990.

78. Synge, J.L. and Schild, A. *Tensor Calculus.* New York: Dover Publications, Inc, 1949.

79. Tarr, Gregory L. *Neural Graphics: An Artificial Neural Network Simulator Users Information Guide.* Unpublished User's Manual. Mar 1991.

80. Teague, Michael R. "Image Analysis via the General Theory of Moments," *J. Opt. Soc. Am., 70/8* :920-930 (Aug 1980).

81. Teh, Cho-Huak and Chin, Roland "Two-Dimensional CAD-Based Object Recognition," *Ninth International Conference on Pattern Recognition.* 382-384. Rome, Nov 1988.

82. Tsai, Roger "Multiframe Image Point Matching and 3-D Surface Reconstruction," *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-5/2* :159 (Mar 1983).

83. Vayda, A.J. and Kak, A.C. "Geometric Reasoning for Pose and Size Estimation of Generic Shaped Object," *Proceedings of the 1990 IEEE International Conference on Robotics and Automation.* 782-789. Cincinnati, OH. May 1990.

84. Wallace, Timothy P. *et al.* "Three-Dimensional Shape Analysis Using Local Shape Descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-3/3* :310-323 (May 1981).

85. Wang, Y.F. *et al.* "Matching Three-Dimensional Objects Using Silhouettes," *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-6/4* :513-518 (Jul 1984).

86. Wells, William III "A Statistical Approach to Model Matching," *Proceedings of the SPIE - The International Society for Optical Engineering, v1381.* 22-29. Bellingham, WA, 1990.